

```
assets/js/app.js document.addEventListener('DOMContentLoaded', function() { const
releasesDropdown = document.getElementById('releases-dropdown'); const btnDownloadZip =
document.getElementById('btn-download-zip'); const filesList = document.getElementById('files-list');
const textEditor = document.getElementById('text-editor'); const tabButtons =
document.querySelectorAll('.tab-btn'); Локальное хранилище для текстов текущего выбранного
релиза,
```

```
// чтобы переключать вкладки мгновенно без повторных запросов к серверу
let currentReleaseTexts = {
  readme: '',
  license: '',
  comment: '',
  log: 'Здесь будет выводиться лог действий (100 строк)...'
};
```

```
/**
 * Отрисовка таблицы файлов в верхнем окне
 * @param {Array} files Массив файлов из JSON
 */
function renderFilesTable(files) {
  if (!files || files.length === 0) {
    filesList.innerHTML = `<tr><td colspan="4" class="text-center
text-muted">В данном релизе нет файлов для скачивания.</td></tr>`;
    return;
  }
```

```
  let html = '';
  files.forEach(file => {
    const icon = file.is_dir ? '📁' : '📄';
    html += `
      <tr>
        <td><span style="margin-right: 8px;">${icon}</span>
        ${escapeHtml(file.name)}</td>
        <td class="text-muted">--</td> <!-- Примечания (коммиты)
прикрутим позже через .meta.json -->
        <td class="text-muted">${file.date}</td>
        <td style="text-align: right;"><span class="text-muted"
style="font-size: 12px;">${file.size}</span></td>
      </tr>
    `;
  });
  filesList.innerHTML = html;
}
```

```
// Вспомогательная функция защиты от XSS при отрисовке имен файлов
function escapeHtml(string) {
  return String(string).replace(/&/g, '&amp;').replace(/</g,
'&lt;').replace(/>/g, '&gt;').replace(/"/g, '&quot;');
}
```

```
// 1. СЛУШАЕМ ВЫБОР РЕЛИЗА В ВЫПАДАЮЩЕМ СПИСКЕ
```

```
releasesDropdown.addEventListener('change', async function() {  
    const selectedRelease = this.value;
```

```
    if (!selectedRelease) {  
        // Если сбросили выбор — возвращаем интерфейс в исходное состояние  
        btnDownloadZip.disabled = true;  
        filesList.innerHTML = `|  | | | |        textEditor.value = 'Текст файла отсутствует или релиз не выбран...';  
        return;  
    }

```

```
    filesList.innerHTML = `|  |  |  |  |
| --- | --- | --- | --- |
|  | | | |

```

```
    // Держаем наш api-модуль  
    const result = await Api.getReleaseData(selectedRelease);
```

```
    if (result.success) {  
        // Активируем кнопку ZIP  
        btnDownloadZip.disabled = false;  
        // Сохраняем тексты в память для вкладок  
        currentReleaseTexts.readme = result.texts.readme || 'Файл README.txt пуст.';  
        currentReleaseTexts.license = result.texts.license || 'Файл LICENSE.txt пуст.';  
        currentReleaseTexts.comment = result.texts.comment || 'Файл COMMENT.txt пуст.';  
        // Отрисовываем файлы в таблице верхнего окна  
        renderFilesTable(result.files);
```

```
        // Автоматически показываем текст той вкладки, которая сейчас активна (по умолчанию README)  
        const activeTab = document.querySelector('.tab-btn.active').getAttribute('data-target');  
        textEditor.value = currentReleaseTexts[activeTab];  
    } else {  
        alert("Ошибка загрузки: " + result.error);  
        filesList.innerHTML = `| >${result.error}</td></tr>`;     } }); | | | |

```

```
// 2. СЛУШАЕМ ПЕРЕКЛЮЧЕНИЕ ВКЛАДОК НИЖНЕГО ОКНА
```

```
tabButtons.forEach(button => {  
    button.addEventListener('click', function() {  
        tabButtons.forEach(btn => btn.classList.remove('active'));  
        this.classList.add('active');  
        const targetTab = this.getAttribute('data-target');  
        // Вытаскиваем текст из памяти без лишних запросов к PHP
```

```
        textEditor.value = currentReleaseTexts[targetTab] || `Текст для
вкладки ${targetTab.toUpperCase()} отсутствует.`;
    });
});
```

```
});
```

From:

<https://wwoss.ru/> - **worldwide open-source software**

Permanent link:

https://wwoss.ru/doku.php?id=tmp_27.06.2026&rev=1782560884

Last update: **2026/06/27 14:48**

