

```
document.addEventListener('DOMContentLoaded', function() {
```

```
    // --- НАДЕЖНОЕ УПРАВЛЕНИЕ МОДАЛКОЙ С ДЕБАГ-МАЯКОМ ---
    // --- УПРАВЛЕНИЕ ЗАГРУЗЧИКОМ ПРОЕКТОВ ---
    const btnCreateProject = document.getElementById('btn-create-project');
    const projectModal = document.getElementById('project-modal');
    const btnModalCancel = document.getElementById('btn-modal-cancel');
    const btnModalSubmit = document.getElementById('btn-modal-submit');
    const newProjectName = document.getElementById('new-project-name');
    const modalErrorBox = document.getElementById('modal-error-box');
    const folderInput = document.getElementById('project-folder-
input');
    const btnSelectFolder = document.getElementById('btn-select-folder');
    const folderStatus = document.getElementById('selected-folder-
status');
```

```
    if (btnCreateProject && projectModal) {
        // Функция проверки условий для активации кнопки "Загрузить"
        function checkInputsValidity() {
            const nameVal = newProjectName.value.trim();
            const hasFiles = folderInput.files && folderInput.files.length >
0;

            if (nameVal !== "" && hasFiles) {
                btnModalSubmit.disabled = false; // Включаем кнопку
                btnModalSubmit.style.opacity = "1";
            } else {
                btnModalSubmit.disabled = true; // Выключаем кнопку
                btnModalSubmit.style.opacity = "0.6";
            }
        }
    }
```

```
    // Следим за вводом имени
    newProjectName.addEventListener('input', checkInputsValidity);
```

```
    // Клик по кастомной кнопке -> триггерим скрытый системный инпут выбора папки
    btnSelectFolder.addEventListener('click', function() {
        folderInput.click();
    });
```

```
    // Следим за тем, что пользователь выбрал папку на ПК
    folderInput.addEventListener('change', function() {
        if (this.files && this.files.length > 0) {
            // Берем имя корневой папки, которую выбрал юзер
            const firstFile = this.files[0];
            const rootFolderName =
firstFile.webkitRelativePath.split('/')[0];
            folderStatus.innerHTML = `Выбрано файлов:
<strong>${this.files.length}</strong><br><span style="font-size:11px;
color:#2da44e;">(из папки: ${rootFolderName})</span>`;
        } else {
            folderStatus.textContent = "Папка не выбрана";
        }
    });
```

```
    }  
    checkInputsValidity();  
});
```

```
// Открытие модалки
```

```
btnCreateProject.addEventListener('click', function() {  
    newProjectName.value = '';  
    folderInput.value = ''; // Сбрасываем выбранные файлы  
    folderStatus.textContent = "Папка не выбрана";  
    if (modalErrorBox) modalErrorBox.classList.add('hidden');  
    btnModalSubmit.disabled = true;  
    btnModalSubmit.style.opacity = "0.6";  
    projectModal.classList.remove('hidden');  
    newProjectName.focus();  
});
```

```
// Отмена
```

```
btnModalCancel.addEventListener('click', function() {  
    projectModal.classList.add('hidden');  
});
```

```
// Нажатие на кнопку "Загрузить" -> отправка терабайтов кода по AJAX
```

```
btnModalSubmit.addEventListener('click', async function() {  
    const projectName = newProjectName.value.trim();  
    if (modalErrorBox) modalErrorBox.classList.add('hidden');
```

```
    const formData = new FormData();  
    formData.append('name', projectName);
```

```
// КРИТИЧЕСКИ ВАЖНО: Пакуем файлы вместе с их путями вложения
```

```
    for (let i = 0; i < folderInput.files.length; i++) {  
        let file = folderInput.files[i];  
        formData.append('project_files[]', file);
```

```
    // Передаем бэкенду относительный путь (например: "my_proj/src/main.php")  
    formData.append('project_paths[]', file.webkitRelativePath);  
    }
```

```
// Меняем статус кнопки на время тяжелой загрузки
```

```
    btnModalSubmit.disabled = true;  
    btnModalSubmit.textContent = "Загрузка...";
```

```
    try {  
        const response = await  
fetch('backend/router.php?action=create_project', {  
            method: 'POST',  
            body: formData  
        });  
        const result = await response.json();  
        if (result.success) {  
            projectModal.classList.add('hidden');
```

```
        btnModalSubmit.textContent = "Загрузить";
        location.reload();
    } else {
        btnModalSubmit.disabled = false;
        btnModalSubmit.textContent = "Загрузить";
        if (modalErrorBox) {
            modalErrorBox.textContent = result.error;
            modalErrorBox.classList.remove('hidden');
        }
    }
} catch (error) {
    btnModalSubmit.disabled = false;
    btnModalSubmit.textContent = "Загрузить";
    alert("Ошибка отправки файлов на сервер: " + error.message);
}
});
}
```

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

Permanent link:
https://wwoss.ru/doku.php?id=tmp_27.06.2026&rev=1782563369

Last update: **2026/06/27 15:29**

