

Подключение CSS и JavaScript файлов в веб-приложении на базе фреймворка Yii2

Андриенко Иван Сергеевич

Приамурский государственный университет имени Шолом-Алейхема

Студент

Аннотация

В данной статье рассматривается процесс подключения CSS и JavaScript файлов в веб-приложении, разработанном с использованием фреймворка Yii2. Описываются методы создания и настройки Asset Bundles, а также их использование для управления ресурсами (CSS, JS) в приложении. Особое внимание уделено подключению внешних библиотек, таких как Font Awesome, и созданию пользовательских скриптов для улучшения функциональности и визуального оформления веб-страниц.

Ключевые слова: Yii2, Asset Bundles, CSS, JavaScript, Font Awesome, веб-разработка, PHP, фреймворк, оптимизация ресурсов.

Connecting CSS and JavaScript files in a web application based on the Yii2 framework

Andrienko Ivan Sergeevich

Sholom-Aleichem Priamursky State University

Student

Abstract

This article discusses the process of connecting CSS and JavaScript files in a web application developed using the Yii2 framework. It describes methods for creating and configuring AssetBundles, as well as their use for managing resources (CSS, JS) in an application. Special attention is paid to connecting external libraries, such as Font Awesome, and creating custom scripts to improve the functionality and visual appearance of web pages.

Keywords: Yii2, Asset Bundles, CSS, JavaScript, Font Awesome, Web development, PHP, framework, resource optimization.

1 Введение

1.1 Актуальность

Современные веб-приложения требуют не только высокой функциональности, но и привлекательного визуального оформления. Для достижения этих целей разработчики активно используют CSS и JavaScript. Фреймворк Yii2 предоставляет мощные инструменты для управления ресурсами, такие как Asset Bundles, которые позволяют эффективно подключать и оптимизировать CSS и JavaScript файлы. Умение работать с

этими инструментами становится важным навыком для разработчиков, так как это позволяет создавать более сложные и производительные веб-приложения.

1.2 Обзор исследований

Разработка веб-приложений и использование современных фреймворков для их создания являются ключевыми направлениями в области информационных технологий. И. В. Дашенко и Е. В. Фешина в своей работе анализируют основные средства разработки веб-ориентированных информационных систем. Особое внимание уделяется инструментам, которые обеспечивают интеграцию внешних файлов и модулей, включая подключение CSS и JavaScript для улучшения интерфейса и функциональности веб-приложений [1]. А. А. Локтионов рассматривает современные фреймворки, включая Yii2, с точки зрения их удобства и возможностей для реализации сложных пользовательских интерфейсов. Автор подчеркивает важность правильного управления подключением ресурсов, таких как стили и скрипты, для обеспечения адаптивности и производительности [2]. О. Ю. Лавлинская и М. С. Бондаренко в своем исследовании проводят обзор технологий разработки веб-приложений, включая интеграцию клиентских технологий, таких как CSS и JavaScript. Авторы отмечают, что использование встроенных механизмов фреймворков, например AssetManager в Yii2, позволяет эффективно организовать подключение и оптимизацию ресурсов [3]. Ю.Р.Поротикова и Л. Н. Амосова акцентируют внимание на технологиях создания веб-сайтов. В своей работе они рассматривают подходы к подключению внешних файлов и их оптимизации для минимизации времени загрузки страниц. Это особенно актуально при разработке на основе PHP-фреймворков, таких как Yii2 [4]. П. А. Козич анализирует использование современных PHP-фреймворков для создания корпоративных веб-порталов. Особое внимание уделяется Yii2 и его инструментам для подключения внешних CSS и JavaScript-файлов. Автор подчеркивает, что правильное управление этими ресурсами позволяет не только улучшить внешний вид, но и повысить производительность приложений [5].

1.3 Цель исследования

Цель исследования – разработка и реализация веб-приложения на базе фреймворка Yii2 с использованием Asset Bundles для управления CSS и JavaScript файлами. В рамках исследования планируется продемонстрировать процесс создания и настройки Asset Bundles, подключения внешних библиотек и создания пользовательских скриптов для улучшения функциональности и визуального оформления веб-страниц.

2 Материалы и методы

Для выполнения работы использовались следующие инструменты и технологии: фреймворк Yii2 для разработки веб-приложения, механизм Asset Bundles для управления CSS и JavaScript файлами, библиотека Font Awesome для подключения иконок и локальный сервер на базе OpenServer для

разработки и тестирования. В процессе работы были созданы Asset Bundles, подключены внешние библиотеки и разработаны пользовательские скрипты для улучшения функциональности и визуального оформления веб-страниц.

3 Результаты и обсуждения

Для управления CSS и JavaScript файлами в Yii2 используется механизм Asset Bundles. Asset Bundle — это класс, который описывает набор ресурсов (CSS, JS, изображения и т.д.), необходимых для работы страницы. В рамках исследования был создан Asset Bundle CustomAsset, который подключает пользовательские CSS и JavaScript файлы.

Класс CustomAsset наследуется от базового класса yii\web\AssetBundle и содержит следующие свойства:

- `$basePath`: указывает путь к корневой директории, где находятся ресурсы. В данном случае используется псевдоним `@webroot`, который указывает на папку web.
- `$baseUrl`: указывает URL, соответствующий `$basePath`. Используется псевдоним `@web`, который соответствует URL корневой директории.
- `$css`: массив, содержащий пути к CSS файлам. В данном случае подключен файл `css/custom.css`.
- `$js`: массив, содержащий пути к JavaScript файлам. В данном случае подключен файл `js/custom.js`.
- `$depends`: массив, содержащий зависимости от других Asset Bundles. В данном случае указана зависимость от `yii\web\YiiAsset`, который предоставляет базовые ресурсы Yii2.

Этот Asset Bundle позволяет централизованно управлять ресурсами, что упрощает их подключение и использование на различных страницах приложения (рис. 1).

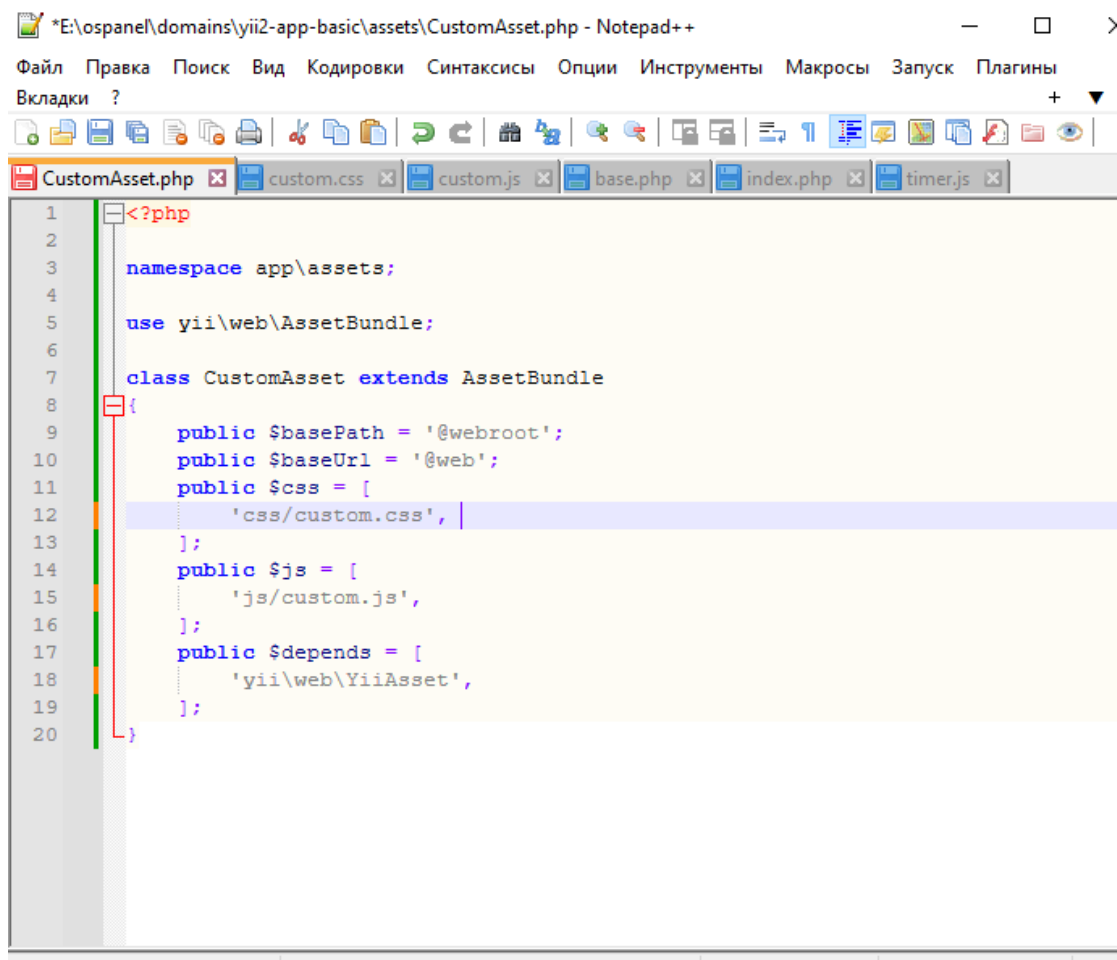


Рис. 1. Код CustomAsset.php

Для подключения созданного Asset Bundle к странице необходимо зарегистрировать его в layout-шаблоне. В файле views/layouts/base.php был добавлен код, который регистрирует CustomAsset вместе с базовым AppAsset. Это позволяет автоматически подключать все CSS и JavaScript файлы, указанные в CustomAsset, на каждой странице приложения.

Регистрация Asset Bundle выполняется с помощью метода register(), который добавляет ссылки на CSS и JavaScript файлы в HTML-код страницы. В результате все ресурсы, описанные в CustomAsset, становятся доступными для использования на странице. Это упрощает процесс управления ресурсами и обеспечивает их корректное подключение (рис. 2).

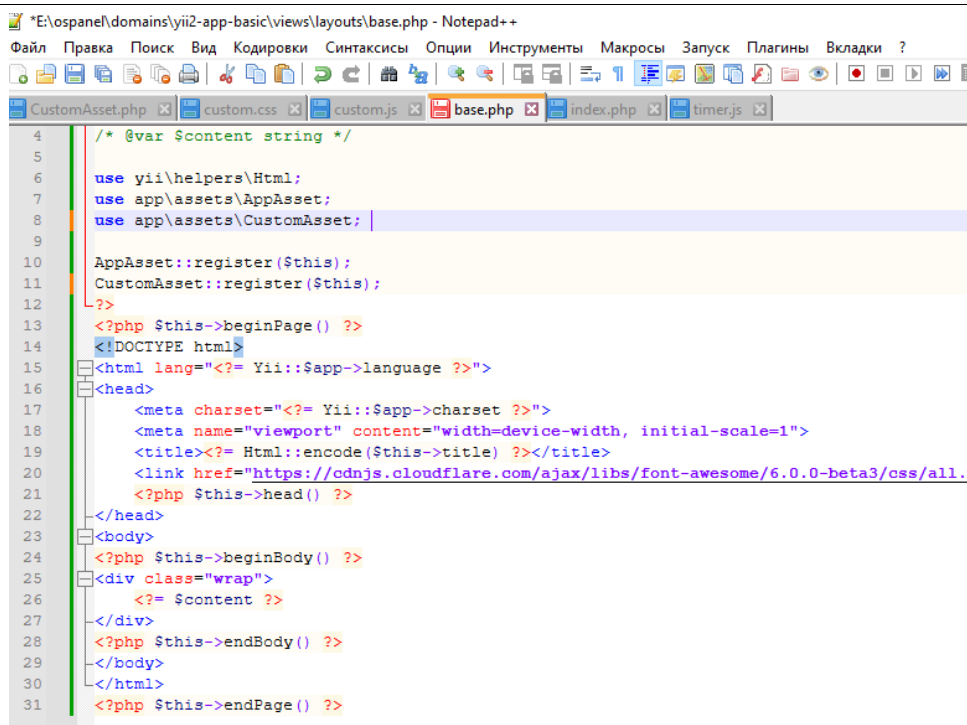


Рис. 2. Регистрация Asset Bundle в layout-шаблоне

Для подключения внешних библиотек, таких как Font Awesome, был использован CDN (Content Delivery Network). В файле views/layouts/base.php добавлена ссылка на библиотеку Font Awesome, что позволяет использовать её иконки на всех страницах приложения. Это делается путём добавления соответствующего тега `<link>` в секцию `<head>` HTML-документа. Использование CDN позволяет загружать библиотеку с внешнего сервера, что уменьшает нагрузку на собственный сервер и ускоряет загрузку страниц. В данном случае Font Awesome используется для добавления иконок, которые улучшают визуальное оформление и удобство интерфейса (рис. 3).

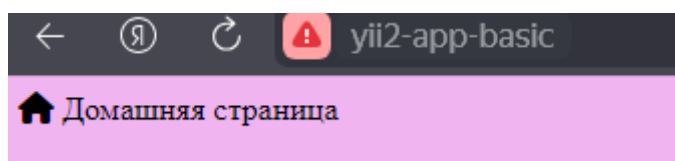


Рис. 3. Подключение внешней библиотеки

Для создания пользовательских скриптов был разработан файл timer.js, который реализует таймер обратного отсчёта. Этот скрипт добавляет интерактивность на страницу, позволяя отображать оставшееся время до определённой даты. Файл timer.js размещён в папке web/js, что делает его доступным для подключения через Asset Bundle или напрямую в представлении (рис. 4).

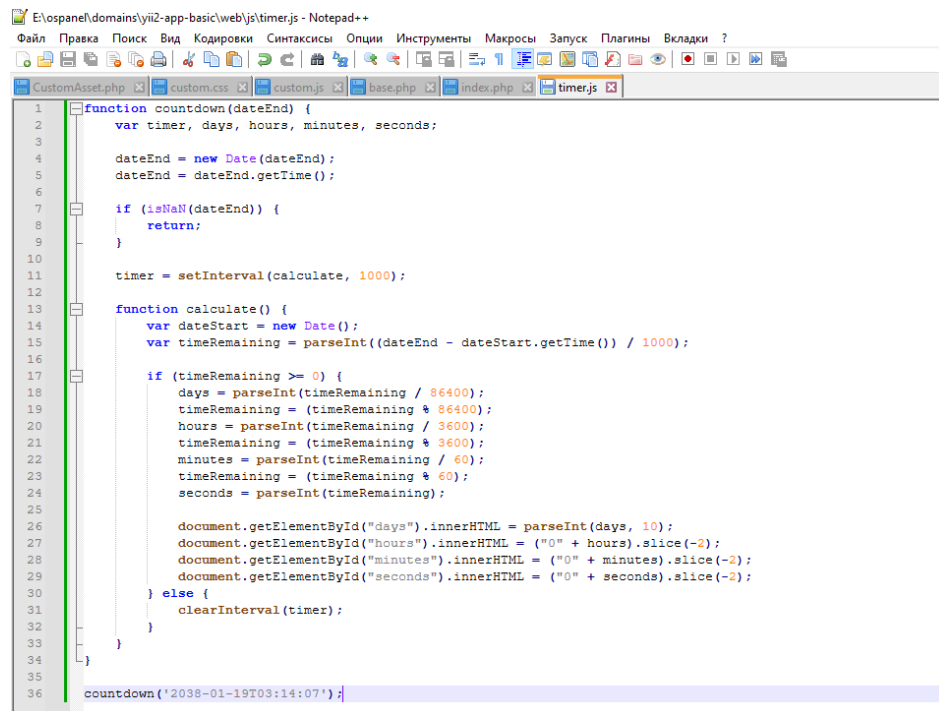


Рис. 4. Реализация таймера

Для подключения пользовательского скрипта на конкретной странице используется метод `registerJsFile()`, который позволяет добавить ссылку на JavaScript файл в HTML-код страницы. В данном случае скрипт `timer.js` подключается в представлении `views/site/index.php`. Это позволяет загружать скрипт только на тех страницах, где он необходим, что оптимизирует загрузку ресурсов (рис. 5).

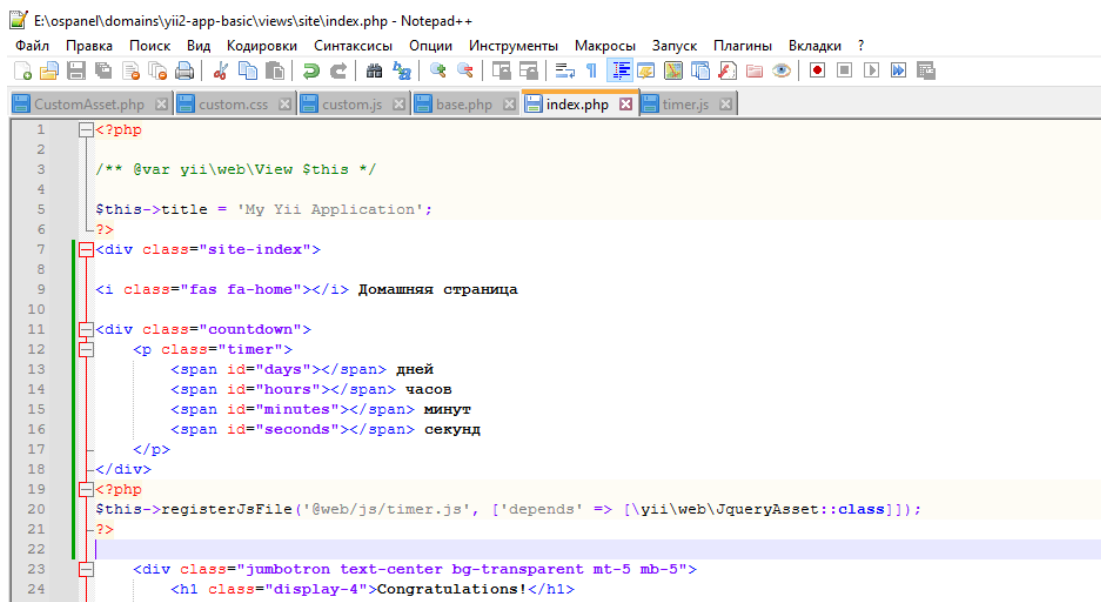


Рис. 5. Подключение таймера

Тестируем таймер. Он обновляется в реальном времени (рис. 6).

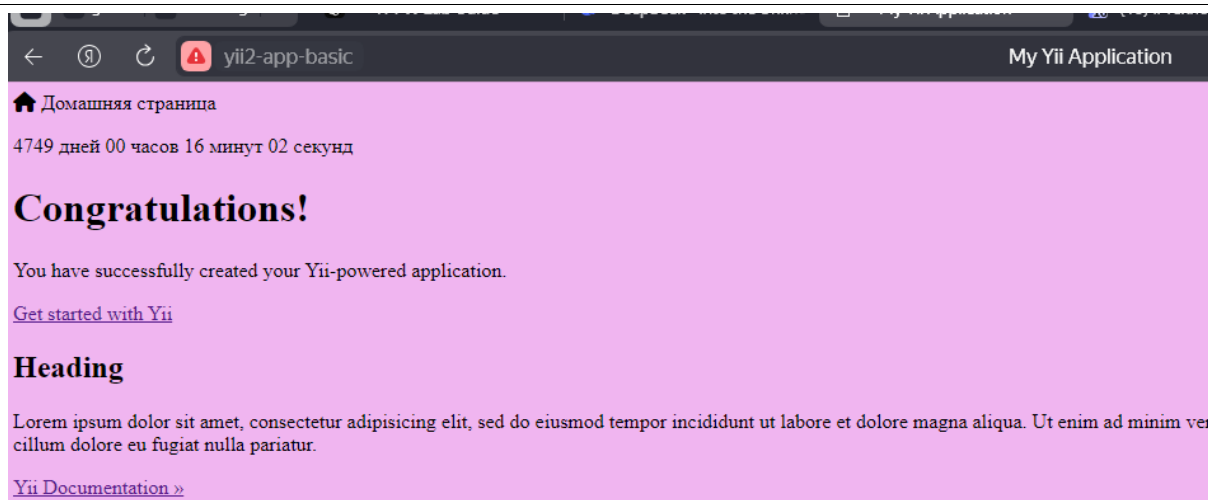


Рис. 6. Работа таймера в реальном времени

Выводы

В ходе выполнения работы было разработано веб-приложение на базе фреймворка Yii2, в котором успешно реализовано управление ресурсами с использованием Asset Bundles. Это позволило организовать централизованное подключение CSS и JavaScript файлов, что значительно упростило процесс разработки и поддержки приложения. Особое внимание было уделено подключению внешних библиотек, таких как Font Awesome, через CDN, что обеспечило быстрое и эффективное использование иконок на всех страницах приложения. Кроме того, был создан пользовательский скрипт `timer.js`, который добавил интерактивность на страницу, демонстрируя возможности Yii2 для работы с пользовательскими JavaScript-файлами.

Результаты работы показали, что Yii2 предоставляет мощные инструменты для управления ресурсами, что делает его удобным выбором для разработки современных веб-приложений. Использование Asset Bundles, внешних библиотек и пользовательских скриптов позволяет создавать гибкие, производительные и визуально привлекательные приложения, что подтверждает актуальность и практическую ценность проведённого исследования.

Библиографический список

1. Дашенко И. В., Фешина Е. В. Средства разработки веб-ориентированных информационных систем // В сборнике: Наука XXI века: проблемы, перспективы и актуальные вопросы развития общества, образования и науки. Материалы XII международной межвузовской научно-практической конференции. Краснодар, 2023. С. 46-52.
2. Локтионов А. А. Нализ современных фреймворков для web-разработки // В сборнике: Цифровые технологии в образовании. Материалы I Международной научно-практической конференции 15–16 декабря 2021 г. 2021. С. 44-48.
3. Лавлинская О. Ю., Бондаренко М. С. Обзор инструментальных средств и

- технологий разработки веб-приложений // Вестник Воронежского института высоких технологий. 2016. № 4 (19). С. 73-78.
4. Поротикова Ю. Р., Амосова Л. Н. Обзор технологий для создания web-сайта // Вестник научного общества студентов, аспирантов и молодых ученых. 2018. № 3. С. 61-68.
 5. Козич П. А. Особенности использования современных PHP-фреймворков для реализации корпоративных веб-порталов // Постулат. 2019. № 1-1 (39). С. 126.