

АВТОМАТИЗАЦИЯ РАЗРАБОТКИ ПОРТАЛОВ С МНОГОКРАТНО ИСПОЛЬЗУЕМЫМИ КОМПОНЕНТАМИ

Навросяк В. Г.

Кафедра интеллектуальных информационных технологий, Белорусский государственный университет информатики и радиоэлектроники Минск, Республика Беларусь
Минск, Республика Беларусь
E-mail: navrosjuk@gmail.com

Данная работа рассматривает принципы разработки веб-порталов. Исследуются возможности применения многократно используемых компонент, их комбинация, а также использование контент-менеджера.

ВВЕДЕНИЕ

В наше время огромную долю IT рынка занимает разработка веб-сайтов. Веб-порталы с многократно используемыми компонентами позволяют упростить процесс создания веб-сайтов, а также реализовать дальнейшую поддержку без вмешательства разработчиков. Веб-портал — сайт в компьютерной сети, который предоставляет пользователю различные приложения, которые работают в рамках этого сайта. Таким образом, создав библиотеку компонент(приложений), разработчик может решать разные бизнес-задачи, не затрачивая много времени на создание веб-сайтов, а собирая сайты из готовых компонент - портлетов.

Портлет — это отдельное небольшое веб-приложение, которое выполняется на портале, портал в свою очередь агрегирует один или несколько портлетов на отдельной веб-странице, которые обычно настраиваются для отдельных пользователей и групп портала. Итог - веб-страница, наполненная несколькими веб-приложениями. Простой пример портлета - корзина покупателя, создав лишь один раз такой портлет, появляется возможность использования данного портлета в других веб-сайтах, где необходим функционал корзины покупок.

Целью этой работы является рассмотрение принципов разработки и использования веб-порталов с многократно используемыми компонентами. Также, так как компоненты могут создаваться разными разработчиками, необходимо реализовать механизм комбинации нескольких портлетов в рамках одной веб-страницы.

I. ЯДРО ПОРТАЛА

Для функционирования веб-приложений и их агрегирования на одной странице необходим такой продукт как портал, в качестве портала был выбран готовый Open Source продукт - Liferay [1]. Liferay - программный продукт, представляющий собой корпоративный портал, то есть решение, предназначенное для централизованного доступа к нескольким различным корпоративным приложениям в одном месте. Liferay иногда описывается как система управле-

ния содержимым (CMS) или платформу для веб-приложений. Написан на языке Java и распространяется под двумя видами лицензий, свободной и проприетарной, используя бизнес-модель двойного лицензирования.

Liferay Portal позволяет пользователям настроить общий доступ к разным приложениям через один-единственный сайт. Это реализуется с помощью функциональных модулей, называемых портлетами. Портлеты должны иметь Backend и Frontend часть так как являются веб-приложениями.

Для автоматизации разработки портлетов необходимо простое решение, которое без вмешательства разработчика будет создавать из серверного и клиентского кода готовое приложение. В качестве языка для Backend разработки используется Java, а также фреймворк OSGi [2]. OSGi - состоит из двух частей. Первая часть представляет собой спецификацию модульных компонентов, называемых пакетами, которые обычно называются надстройками. Спецификация определяет инфраструктуру для жизненного цикла пакета и определяет, как пакеты будут взаимодействовать. Вторая часть OSGi - это реестр служб уровня виртуальной машины Java (JVM), который пакеты могут использовать для публикации, обнаружения и привязки к службам в сервис-ориентированной архитектуре (SOA) [3].

В результате получатся контроллеры, являющиеся промежуточной частью между клиентской частью и сторонним API или базой данных. Frontend часть реализуется на языке Angular - платформа веб-приложений с открытым исходным кодом на основе TypeScript. Результат компиляции Frontend кода - html файлы, отвечающие за отображение и js файлы, отвечающие за логику веб-приложения и обработку событий.

Был реализован фреймворк, который собирает результат компиляции Frontend и Backend кода, как единый портлет. Он основан на технологии JSP - технология, позволяющая веб-разработчикам создавать содержимое, которое имеет как статические, так и динамические ком-

поненты. Архитектура портлета представлена на рис.1.

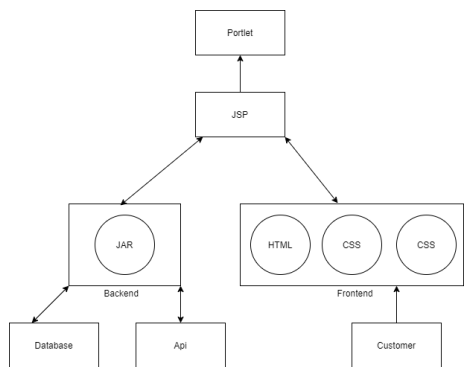


Рис. 1 – Архитектура разработки портлета

II. КОМБИНАЦИЯ ПОРТЛЕТОВ НА ВЕБ-СТРАНИЦЕ

Подразумевается, что разработчики будут реализовывать портлеты, добавлять их в общую библиотеку, после чего пользователи комбинируют портлеты из библиотеки составят свои веб-сайты. Однако, так как разные разработчики занимаются реализацией, у веб-приложений нету общего стиля и при добавлении разных портлетов, веб-страница не будет выглядеть как единое целое. Поэтому, был реализован функционал, позволяющий менять конфигурацию портлета в зависимости от содержимого веб-страницы.

При разработке портлета, необходимо выполнить конфигурацию портлета - заполнить XML файл, который регулирует цвет, шрифт, размер в зависимости от других компонент на странице. На рис.2 представлен пример веб-страницы, состоящей из трех компонент: график, статистика, описание. В левой части рис.2 видно как выглядела веб-страница до выполнения конфигурации портлетов, компоненты имеют разные цвета фона, размеры шрифта, веб-страница не выглядит как единое целое. После в файле конфигурации был настроен основной цвет веб-страницы (белый) и размеры шрифтов, компоненты изменили свой стиль, так как при их создании был реализован конфигурационный файл, и в правой части рис.2 можно видеть конечный

результат, который выглядит как единая веб-страница.

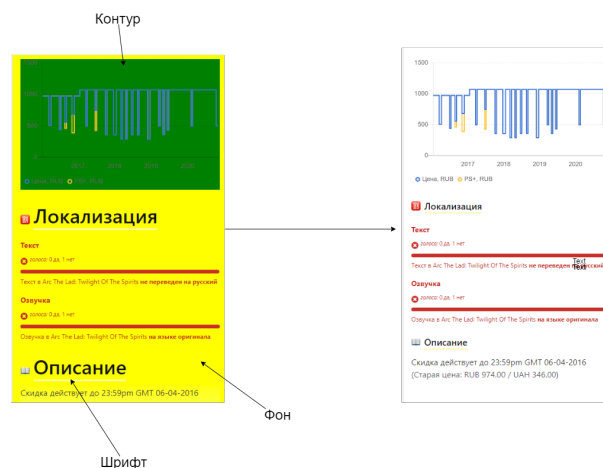


Рис. 2 – Пример конфигурации портлета

III. ЗАКЛЮЧЕНИЕ

Разработка веб-сайтов для решения бизнес-задач является финансово затратным процессом. Для упрощения реализации и сокращения затрат в данной работе предложен подход, который позволит пользователям создавать свои веб-сайты из разных веб-компонент при этом не требуя знания языков программирования.

В работе рассмотрена реализация веб-порталов с многократно используемыми компонентами. Описана технология создания портлетов и комбинации нескольких портлетов на веб-странице. Дальнейшее направление исследований включает возможность комбинации портлетов без вмешательства разработчика и пользователя, а также внедрения контент-менеджера для управления и версионирования контента веб-портала.

СПИСОК ЛИТЕРАТУРЫ

1. Liferay Portal Systems Development : Build Dynamic, Content-rich, and Social Systems on Top of Liferay / X. Y. Jonas . – Packt Publishing Limited, 2016. – 317 с.
2. Osgi in Action: Creating Modular Applications in Java / R. I. Hall . – Manning Publications, 2011. – 375 с.
3. OSGI [Электронный ресурс] / OSGI Alliance Community. – Режим доступа: <https://www.osgi.org/>. – Дата доступа: 12.05.2020.