

РАЗРАБОТКА ВЕБ-ПРИЛОЖЕНИЯ ДЛЯ УПРАВЛЕНИЯ СКЛАДСКИМИ ПРОЦЕССАМИ ИНТЕРНЕТ-МАГАЗИНОВ

*В.Г.Ротарь,
А.Е.Трофимова, М.Ткачев,
Томский политехнический университет
E-mail: aet11@tpu.ru*

Введение

Электронная коммерция становится популярной во многих сферах предпринимательства. По данным статистики PayPal около 67% компаний отмечают важность электронной коммерции в успешном ведении и росте их бизнеса. Торговля через онлайн сервисы позволяет предпринимателю снизить затраты на содержание торговых и складских площадей [1].

На сегодняшний день является актуальным создание отдельной системы управления складом с возможностью синхронизации с интернет-магазином, созданным на любой платформе. Реализация системы управления складскими операциями в виде веб-приложения позволяет реализовать все его преимущества, такие как кроссплатформенность, доступность с различных устройств и отсутствие необходимости установки. Независимость веб-приложения от платформы интернет-магазина позволяет осуществлять более простой межплатформенный переход в случае необходимости [2].

Целью проекта является разработка веб-приложения с функционалом синхронизации интернет-магазина и с системой управления складскими процессами, учетом продукции, а также основных и сопутствующих расходов интернет-магазина.

Архитектура веб-приложения

Для разработки приложения была выбрана одна из самых перспективных технологий на основе JavaScript, а именно, стек MEAN (MongoDB, Express.JS, Angular2 и Node.JS). Стек MEAN охватывает полный цикл проекта от разработки на стороне клиента (Angular2) до разработки на стороне сервера (Node.JS, Express), при этом в качестве базы данных использована MongoDB [3].

В разрабатываемом веб-приложении реализована следующая логика взаимодействия пользователя с системой. Клиенту приложения предоставляется возможность добавления продуктов, заказанные у поставщиков, путем импорта из файла excel-формата. Таким образом, создается отчет о заказе данного ассортимента продукции с соответствующими параметрами. После формирования заказа генерируется запрашиваемая продукция и создаются отчеты расходов. Пользователь может создавать записи расходов различных категорий. Как только произойдет приемка товаров на складе, визуализируется рекомендуемое на основании статистики продаж местоположение товара в складском помещении, с дальнейшим подтверждением местоположения продукта вручную либо автоматически с помощью RFID-меток. Также пользователь системы управления складом может получить доступ к финансовым статистическим данным бюджета за выбранный период и к ресурсам компании. Реализована возможность получения рекомендации для последующего заказа продукции на основании статистики продаж за аналогичный период.

Особенностью проектируемого веб-приложения является его REST API (Representational State Transfer Application programming interface), который позволяет создать набор ссылок, по которым отправляется информация от интернет-магазина во время покупки товара, тем самым синхронизируя операции интернет-магазина и приложения складского учета. За счет использования принципов REST API приложение не завязано от технологии, которая используется на стороне интернет-магазина, а также особо удобна при смене платформы интернет-магазина. Пользователь в этом случае не испытывает особых трудностей в настройке новой системы управления складом [4].

Архитектура серверной части веб-приложения для формирования ответа клиентским запросам реализована с использованием паттерна MVC (Model-View-Controller), показано на рисунке 1. Обработка запроса с клиентской части к серверу приложения происходит при маршрутизации через определенный контроллер. В процессе обработки контроллер отправляет запрос к базе данных через модели и, получив информацию, формирует ответ, который потом от контроллера направляется на интерфейсную, браузерную часть приложения. Ответ представляет собой JSON-файл, используемый для отображения данных на клиентской части приложения [5].

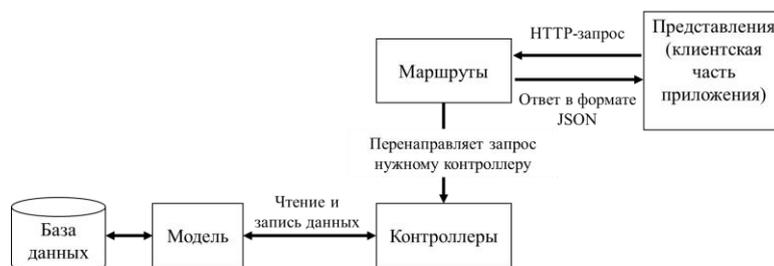


Рис. 1. Схема реализации архитектуры MVC

Проектирование базы данных приложения

Проектирование модели базы данных представляет собой объявление сущности и создание ее основных полей. База данных приложения включает в себя следующие сущности: *Шаблон*, *Продукт*, *Образец продукта*, *Вендор*, *Заказ*, *Бренд продукта*, *Категорию продукта*, *Чек* и *Расход*. На рисунке 2 диаграммы отношений базы данных в нотации Мартина показаны сущности базы данных вместе с их полями, а также определены связи вида 1:1 и 1:M. Ключами у всех сущностей объявлены уникальные идентификаторы `_id`, автоматически генерируемые самой базой данных MongoDB [6].

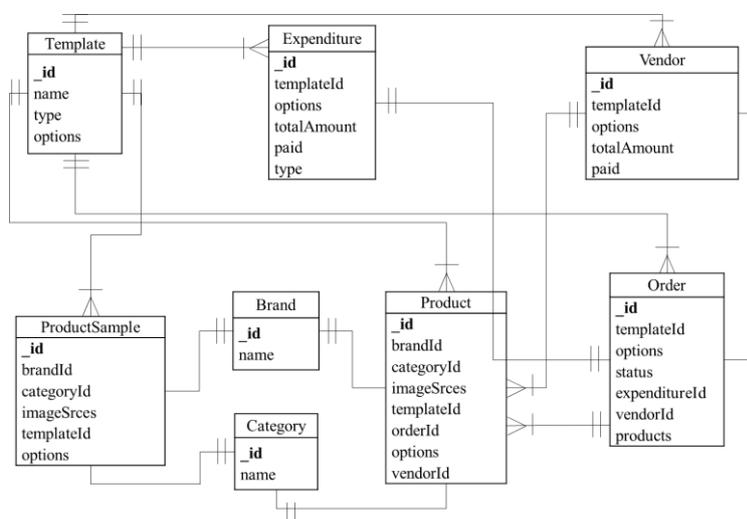


Рис. 2. Диаграмма отношений сущностей базы данных в нотации Мартина

Заключение

В результате разработано веб-приложение для управления складскими процессами и расходами. Данная система позволяет вести учет продукции и визуально отображать расположение товаров в складском помещении компании. Веб-приложение отображает информацию о финансовых статистических данных бюджета и складской ассортимент товаров интернет-магазина. Используемый для создания приложения современный стек технологий MEAN эффективен функционально, что позволило создать систему управления складом с возможностью синхронизации с интернет-магазином, разработанным на любой платформе.

Список использованных источников

1. Покровская Л. Л., Копачев А. А. Электронная коммерция в сфере информационных услуг. – Directmedia, 2015. – 173 с.
2. Макаров В. В., Колотов Ю. О. Развитие интернет-коммерции //Экономический анализ: теория и практика. – 2009. – №. 26. – 60-64 с.
3. Бойко А. М., Войнич С. Г. Использование стека MEAN для создания современных веб-приложений //Цифровая трансформация. – 2017. – №. 1. – 11-15 с
4. Зотова Ю. А., Котилевец И. Д. Разработка архитектуры REST API для взаимодействия с сервисами приложения //Информационные технологии и математическое моделирование систем. – 2018. – 61-65 с.
5. Симонова О. Н., Лясин Д. Н. Шаблон проектирования MVC как эффективное средство построения архитектуры программной системы //Современные наукоемкие технологии. – 2014. – №. 5-2. – 96-97 с
6. Dayley V., Dayley V., Dayley C. Node. js, MongoDB and Angular Web Development: The definitive guide to using the MEAN stack to build web applications. – Addison-Wesley Professional, 2017. – 638 с.