

МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РЕСПУБЛИКИ КАЗАХСТАН

СӘТБАЕВ УНИВЕРСИТЕТІ

Институт информационных и телекоммуникационных технологий

Кафедра " Компьютерная и программная инженерия "


ДОПУЩЕН К ЗАЩИТЕ

Заведующий кафедрой ПИ

канд. техн. наук, доцент,

ассистент-профессор

 Р. Юнусов

" 16 "  2019 г.

ПОЯСНИТЕЛЬНАЯ ЗАПИСКА

к дипломному проекту

На тему: " Разработка web-приложения для интернет-торговли с интеграцией программного продукта 1С: Предприятие."

по специальности 5В070400 – Вычислительная техника и программное обеспечение


Выполнил

Куспаков Н.А.

Научный руководитель

лектор

 Г.А.Омарова

" 16 "  2019 г.

Алматы 2019

МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РЕСПУБЛИКИ КАЗАХСТАН

СӘТБАЕВ УНИВЕРСИТЕТІ

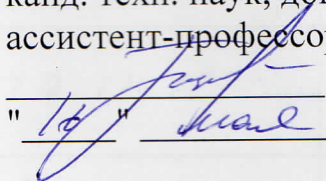

Институт информационных и телекоммуникационных технологий

Кафедра "Компьютерная и программная инженерия"

5B070400 – Вычислительная техника и программное обеспечение

УТВЕРЖДАЮ

Заведующий кафедрой ПИ
канд. техн. наук, доцент,
ассистент-профессор

 Р. Юнусов
" 16 "  2019 г.

ЗАДАНИЕ

на выполнение дипломного проекта

Обучающемуся Кусакову Нугману Алимхановичу

Тема: Разработка web-приложения для интернет-торговли с интеграцией программного продукта 1С: Предприятие.

Утверждена приказом проректора по академической работе № 1162 - б
от "16" октября 2018 г.
"17" мая 2019 г.

Срок сдачи законченного проекта

Исходные данные к дипломному проекту: Техническое задание, описание необходимых функций проекта.

Перечень подлежащих разработке в дипломном проекте вопросов:

а) реализация функциональных возможностей модуля интеграции 1С: Предприятие и интернет-магазина;

б) проектирование и разработка пользовательского интерфейса модуля;

в) разработка, отладка, тестирование программного комплекса.

Перечень графического материала (с точным указанием обязательных чертежей): представлены 23 рисунка.

Рекомендуемая основная литература: из 4 наименований.

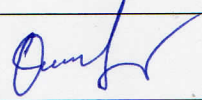

ГРАФИК

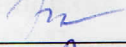
подготовки дипломного проекта

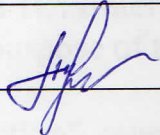
Наименование разделов, перечень разрабатываемых вопросов	Сроки представления научному руководителю и консультантам	Примечание
1. Анализ предметной области, разработка технического задания	17.10.18	—
2. Проектирование механизма модуля интеграции 1С: Предприятие и его интерфейса	1.11.18	—
3. Разработка дизайна интерфейса модуля интеграции 1С: Предприятие, реализация контроллеров	27.11.18	—
4. Тестирование модуля интеграции 1С: Предприятие	8.03.19	—
5. Написание пояснительной записки к дипломному проекту	6.09.19	—

Подписи

консультантов и нормоконтролера на законченный дипломный проект с указанием относящихся к ним разделов проекта

Наименования разделов	Консультанты, И.О.Ф. (уч. степень, звание)	Дата подписания	Подпись
Программное обеспечение	З.М.Өмірбекова Ассистент	6.06.19	
Нормоконтролер	Г.А.Омарова Лектор	6.05.19	

Научный руководитель _____  Г.А.Омарова

Задание принял к исполнению обучающийся _____  Куспаков Н.А.

Дата _____ "11" сентября 2019 г.

ОТЗЫВ

НАУЧНОГО РУКОВОДИТЕЛЯ

на дипломный проект
(наименование вида работы)
Куспакова Нугмана
(Ф.И.О. обучающегося)
5B070400 – Вычислительная техника и программное обеспечение
(шифр и наименование специальности)

Тема: «Разработка web-приложения для интернет-торговли с интеграцией программного продукта 1С: Предприятие»

Данный дипломный проект выполнен на достаточно актуальную тему. К настоящему времени в Казахстане зарегистрировано около 4000 интернет-магазинов, и большая часть из них — это небольшие интернет-магазины в которых менее 1000 наименований товаров. В странах СНГ самым популярным ПО для ведения учета по товарам, его остаткам, его ценообразованием является учетная система 1С: Предприятие: Управление торговлей. И тут встает вопрос интеграции данных программных продуктов в виде интернет магазина и 1С:Предприятие Управление торговлей. Дипломный проект состоит из 4-х разделов, включая введение, заключение и приложений.

Первый раздел работы "Исследовательский раздел" раскрывает цель разработки модуля, а также приводятся сравнения аналогов, их функций, преимуществах и недостатках.

Во втором разделе обзревается технология реализации данного модуля интеграции, а также инструменты, выбранные для его создания.

В третьем разделе была создана проектная документация, включающая в себя описание архитектуры системы и самого модуля, ER диаграмма и диаграмма взаимодействия и их описание, а также рисунки создания пользовательского интерфейса.

Четвертый раздел – “Экспериментальный раздел”. В данном разделе был описан функционал модуля Интеграции, и то как данные функции были реализованы.

В целом данный дипломный проект выполнен с учетом всех требований, предъявляемых к дипломному проекту по специальности 5B070400 - «Вычислительная техника и программное обеспечение», студент Куспаков Н. рекомендован к защите дипломного проекта и заслуживает присвоения академической степени «бакалавра» по специальности 5B070400 - «Вычислительная техника и программное обеспечение».

Научный руководитель

лектор

(должность, уч. степень, звание)

Омарова Г. А.

(подпись)

« 16 » _____ 2019г.

Протокол анализа Отчета подобия Научным руководителем

Заявляю, что я ознакомился(-ась) с Полным отчетом подобия, который был сгенерирован Системой выявления и предотвращения плагиата в отношении работы:

Автор: Куспаков Нугман

Название: диплом Куспаков Нугман

Координатор: Гульнара Омарова

Коэффициент подобия 1: 19,5

Коэффициент подобия 2: 5,4

Тревога: 2

После анализа Отчета подобия констатирую следующее:

- обнаруженные в работе заимствования являются добросовестными и не обладают признаками плагиата. В связи с чем, признаю работу самостоятельной и допускаю ее к защите;
- обнаруженные в работе заимствования не обладают признаками плагиата, но их чрезмерное количество вызывает сомнения в отношении ценности работы по существу и отсутствием самостоятельности ее автора. В связи с чем, работа должна быть вновь отредактирована с целью ограничения заимствований;
- обнаруженные в работе заимствования являются недобросовестными и обладают признаками плагиата, или в ней содержатся преднамеренные искажения текста, указывающие на попытки сокрытия недобросовестных заимствований. В связи с чем, не допускаю работу к защите.

Обоснование:

.....
.....
.....
.....
.....
.....

16.05.19.
.....

Дата

.....
.....

Подпись Научного руководителя

АННОТАЦИЯ

Дипломная работа выполнена на шестидесяти пяти страницах, состоит из введения, четырех разделов, заключения, списка литературы и нескольких приложений. Дипломный проект является модулем интеграции интернет-магазина и УС 1С: Предприятие.

Во введении дипломной работы рассказывается о цели проекта и выполненных задачах.

Первый раздел работы "Исследовательский раздел" раскрывает цель разработки модуля, а также приводятся сравнения аналогов, их функций, преимуществах и недостатках.

Во втором разделе обзревается технология реализации данного модуля интеграции, а также инструменты, выбранные для его создания.

В третьем разделе была создана проектная документация, включающая в себя описание архитектуры системы и самого модуля, ER диаграмма и диаграмма взаимодействия и их описание, а также рисунки создания пользовательского интерфейса.

Четвертый раздел – "Экспериментальный раздел". В данном разделе был описан функционал модуля Интеграции, и то как данные функции были реализованы.

Итогом данного дипломного проекта стал модуль способный в автоматическом режиме производить выгрузку номенклатуры, ее описания и изображения.

ANNOTATION

The thesis has done on sixty-five pages. It consists of an introduction, four sections, a conclusion, a list of references and several annexes. The graduation project is a module for the integration of an online store and 1С: Enterprise.

In the introduction of the thesis describes the purpose of the project and the tasks performed.

The first section of the work "Research Section" reveals the purpose of the module development, as well as comparisons of analogues, their functions, advantages and disadvantages.

The second section reviews the implementation technology of this integration module, as well as the tools selected to create it.

In the third section, the project documentation was created, which includes a description of the system architecture and the module itself, an ER diagram and interaction diagram and their description, as well as drawings for creating a user interface.

The fourth section is "Experimental Section". In this section, the functionality

of the Integration module was described, and how these functions were implemented.

The result of this graduation project is a module capable of automatically unloading the nomenclature, its descriptions and images.

АНДАТПА

Диссертациялық жұмыс алпыс бес беттен тұрады, ол кіріспеден, төрт бөлімнен, қорытындыдан, әдебиеттер тізімі мен бірнеше қосымшалардан тұрады. Дипломдық жоба - онлайн-дүкен мен 1С: Кәсіпорын интеграциялау модулі.

Диссертацияда жобаның мақсаты мен орындалатын тапсырмалар сипатталады.

«Зерттеу бөлімі» жұмысының бірінші бөлімінде модульді дамытудың мақсаты, сондай-ақ аналогтардың салыстыру, олардың функциялары, артықшылықтары мен кемшіліктері айқындалады.

Екінші бөлікте бұл интеграциялық модульді енгізу технологиясы, сондай-ақ оны құруға арналған құралдар қарастырылады.

Үшінші бөлімде жүйелік архитектураның сипаттамасы және модульдің өзі, ER диаграммасы және өзара әрекеттесу схемасы және олардың сипаттамасы, сондай-ақ пайдаланушы интерфейсін жасау үшін сызбалар бар жобалық құжаттама жасалды.

Төртінші бөлім - «Тәжірибелік бөлім». Бұл бөлімде интегралдау модулінің функционалдылығы сипатталды және осы функциялар қалай орындалды.

Бұл бітіру жобасының нәтижесі номенклатураны, оның сипаттамаларын және суреттерін автоматты түрде түсіруге қабілетті модуль.

СОДЕРЖАНИЕ

	Введение	9
1	Исследовательский раздел	10
1.1	Цель разработки модуля	10
1.2	Термины и сокращения	11
1.3	Существующие аналоги	11
1.3.1	Обзор CMS Bitrix	12
1.3.2	Обзор CMS OpenCart	13
1.3.2	Обзор модулей	13
2	Технологический раздел	15
2.1	Архитектурный паттерн MVC-L	15
2.1.1	Способ разработки веб приложения	15
2.1.2	Основная структура каталогов	15
2.1.3	Controller (Контроллер)	16
2.1.3.1	Файл контроллера	16
2.1.3.2	Доступ через URL	16
2.1.4	View (Представление)	17
2.1.5	Model (Модель)	17
2.1.6	Language (Язык)	17
2.1.7	Функциональность внешнего интерфейса	18
2.2	Преимущества используемых технологий	18
2.2.1	Преимущества PHP	18
2.2.2	PHP Storm	19
2.3	Стандарт обмена CommerceML	20
3	Проектный раздел	22
3.1	Архитектура системы	22
3.2	Описание диаграммы вариантов использования	22
3.3	Разработка ER диаграммы	24
3.4	Проектирование интерфейса	24
3.4.1	Создание концепции	25
3.4.2	Создание прототипа	25
4	Экспериментальный раздел	27
4.1	Функции модуля интеграции 1С: Предприятие	27
4.2	Выгрузка из 1С: Предприятие	27
4.3	Загрузка данных из созданного 1С: Предприятие файла в базу интернет магазина	28
4.4	Автоматизация выгрузки данных	29
4.5	Алгоритм выгрузки данных на сайт	30
4.5.1	Авторизация на сайте	31
4.5.2	Инициализация на сайте	33
4.5.3	Выгрузка файлов на сайт	33

4.5.4	Ожидание окончания обработки данных на сайте	34
4.5.5	Распаковка изображений	36
4.6	Добавление номенклатуры в 1С: Предприятие	37
4.7	Тестирование работы модуля интеграции 1С: Предприятие	38
	Заключение	40
	Список использованной литературы	41
	Приложение А. Техническое задание	42
	Приложение Б. Текст программы	45

ВВЕДЕНИЕ

В современном мире многие компании, работающие в режиме offline, в какой-то момент понимают, что им необходимо расширяться и выходить в перспективную область online продаж или предоставления информации, в частности в Интернет. Долгие годы интернет является не только средством для предоставления медиа-контента, но и эффективной площадкой для ведения продаж и своего бизнеса, в данном случае речь идет об интерне-магазинах.

К настоящему времени в Казахстане зарегистрировано около 4000 интернет-магазинов, и большая часть из них — это небольшие интернет-магазины в которых менее 1000 наименований товаров. Но даже с такими объемами продукции сложно вести учет в неспециализированном программном обеспечении. В странах СНГ самым популярным ПО для ведения учета по товарам, его остаткам, его ценообразованием является учетная система 1С: Предприятие: Управление торговлей.

И у компаний становится на повестку следующий вопрос - каким образом имеющуюся базу данных, которую ведет предприятие, передавать на сайт как можно чаще и проще, не теряя её актуальности. Именно в таких вопросах используется, в основном, интеграция. Каждая такая интеграция позволяет производить быструю и надёжную передачу информации из одной программной среды в другую.

В данном дипломном проекте именно такой интеграции и уделено все внимание. Интеграция осуществляется с помощью модуля, написанного на языке PHP на стороне интернет-магазина и УС «1С: Предприятие».

Целью данного дипломного проекта является не создание типового интернет магазина, а успешная интеграция «1С: Предприятие: Управление торговлей» в его структуру. Поэтому за основу интернет магазина был взят CMS OpenCart, так как он идеально подходит по параметрам количества наименований и пользуется популярностью в среднем и малом сегменте интернет-магазинов.

Цель дипломного проекта – реализовать интернет-магазин с модулем интеграции данных о заказах и номенклатуре.

В соответствии с поставленной целью были определены задачи дипломного проекта:

- проанализировать документацию и интернет-источники по теме дипломного проекта;
- систематизировать и отобрать материал;
- реализовать интернет-магазин с использованием выбранной CMS;
- внедрить средства интеграции данных с интернет-магазином для обмена информации о заказах и номенклатуре с конфигурацией на платформе «1С: Предприятие:»;

1 Исследовательский раздел

1.1 Цель разработки модуля

Данный дипломный проект представляет собой интернет-магазин на базе OpenCart с модулем интеграции 1С: Предприятие: Предприятие 8.3. Главным преимуществом данного проекта является большая распространенность платформы OpenCart и его открытый исходный код, а также отсутствие рисков, связанных с лишением поддержки со стороны 1С: Предприятие из-за внесения изменений в конфигурацию системы. Так же неоспоримым плюсом является финансовая составляющая, т.к. OpenCart является свободным программным обеспечением и базируется на скриптовом языке общего назначения, интенсивно применяемого для разработки веб-приложений – PHP.

В возможности модуля входит:

- Выгрузка полной иерархии категорий
- Выгрузка изображений
- Выгрузка описаний товаров
- Выгрузка цен номенклатуры

Для использования модуля в купе с интернет-магазином необходимо интернет соединение, хостинг с размещенным на нем интернет-магазином, а реализация должна происходить с помощью встроенного модуля 1С: Предприятие для обмена с Web-сайтами (рисунок 1.1).

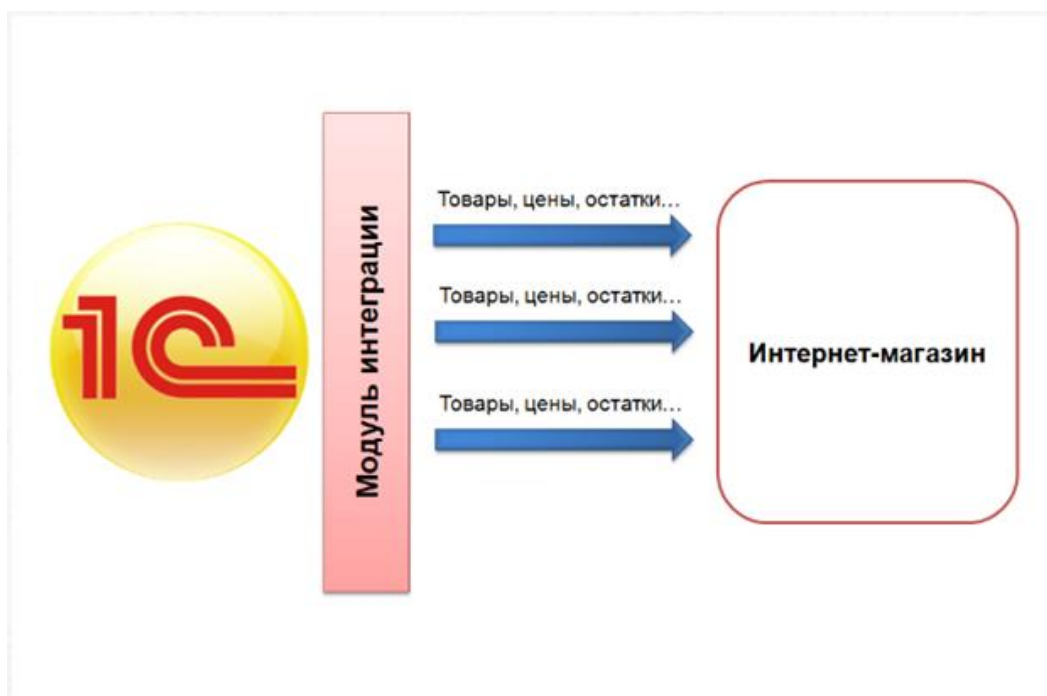


Рисунок 1.1 – Алгоритм работы модулей синхронизации

1.2 Термины и сокращения

Термины и сокращения, использованные при разработке, а также специфические термины, связанные с реализацией и используемыми технологиями отображены в таблице 1.

Таблица 1 – Термины , сокращения, и их определения

Сокращение или термин	Определение
БД	База данных
ОС	Операционная система
API	Application Programming Interface
УС	Учетная система
CMS	Content Management System
HTTP	HyperText Transfer Protocol
SQL	Structured Query Language
UML	Unified Modeling Language
URL	Uniform Resource Locator
SQL	Structured query language
MVCL	Model- View- Controller-Language

1.3 Существующие аналоги

Существует огромное количество CMS: Content Management System или система управления содержимым (рисунок 1.2). Некоторые системы ориентированы только на решение конкретных задач (блоги, интернет-магазины, форумы), другие универсальны и предоставляют разработчику удобную среду проектирования и программирования для разработки чего угодно. Часть CMS состоит из набора функциональных блоков и модулей, другие являются монолитными, неделимыми и даже зашифрованными. Некоторые системы поставляются бесплатно и с возможностью внесения собственных модификаций, а некоторые предоставляются за деньги и не дают возможности редактирования ядра. Рассмотрим самые популярные системы.

НАЗВАНИЕ	ПРОЕКТЫ	ПАРТНЕРЫ	ОЦЕНКА ПОЛЬЗОВАТЕЛЕЙ	СРАВНИТЬ
1С-Битрикс	6 740 проектов	1 303 партнера	4.4	☆
Opencart	1 687 проектов	484 партнера	4.6	☆
WordPress	867 проектов	417 партнеров	4.5	☆
Joomla!	906 проектов	354 партнера	4.5	☆
MODX	633 проекта	282 партнера	4.7	☆

Рисунок 1.2 – Распространенность популярных CMS

1.3.1 Обзор CMS Bitrix

Плюсы:

Распространенность, позволяющая без проблем найти разработчика в любое время и не привязываться к одному или двум специалистам;

Благодаря очень тесной интеграции с 1С: Предприятие, система оттачивается под конкретные бизнес-процессы: автоматизированное создание счетов-фактур, загрузка отчетов для бухгалтерии, мини-документооборот в соответствии со стандартами учета. Но! Интеграция работает только при использовании коробочных версий учета без изменений, что крайне редко. В противном случае вам будет необходимо потратить деньги на дополнительную доработку механизма интеграции, которая по цене будет соизмерима с аналогичной работой на других CMS;

Удобная панель управления для администрирования интернет-магазина.

Недостатки:

Платная, стоимость лицензии значительная - около 170 000 тг. (для интернет-магазинов это издание для малого бизнеса);

Система требует вычислительных ресурсов, для нормальной работы нам необходим мощный процессор и большой объем оперативной памяти. Некоторые хостинговые компании даже имеют специальные тарифы для сайтов на базе Битрикс;

Масштабируется дорого, архитектура содержит много старого кода, который передается от версии к версии, разработчик «Битрикс» занимает в среднем на час заметно больше, чем разработчики, специализирующиеся на других CMS.

1.3.2 Обзор CMS OpenCart

Плюсы:

Система является международной и исключительно магазинной, с бесплатным открытым кодом и лицензией;

В сравнении с конкурентами в сегменте электронной коммерции OpenCart имеет более высокую производительность, магазин с 20-30 тысячами товаров может функционировать даже на дешевом хостинге;

Качественная архитектура и схема разделения данных приложения MVC. MVC предоставляет собой быстрый и комфортный способ работы с кодом. Разработчик уделяет огромное внимание удобству читаемости кода, в репозитории постоянно обновляются комментарии по переименованию переменных, исправлению отступов и т.д.

Недостатки:

– есть определенные проблемы с SEO-оптимизацией.

Заключение:

«Битрикс» - система промышленного класса. Поэтому, пока вы не настроите CMS под свои нужды, управлять проектом довольно сложно. Проекты по этой системе выполняются довольно долго и стоят значительно больше. Он лучше подходит для крупных предприятий, где требуется тесная бесшовная интеграция с 1С: Предприятие и по старинке иметь своего рода «условную» безопасность в виде возможности прийти в офис Битрикс, чтобы пожаловаться на возникшие ошибки. Средняя цена качественного проекта составляет около 3 000 000 тг.

«Opencart» - это наиболее оптимальное решение для малого и среднего бизнеса, где важны прежде всего цена, быстрый запуск, скорость работы и минимальные затраты на обслуживание и развитие сайта. Средняя цена качественного проекта составляет около 1 000 000 тг.

1.3.3 Обзор модулей

Web-студий которые занимаются интеграцией 1С: Предприятие именно с OpenCart на территории Казахстана посредством интернета найти не удалось, как и модуля отечественной разработки. На территории СНГ посредством сети интернет были найдены аналоги со схожим функционалом.

Модуль интеграции NeoSeo – стоимость интеграции под ключ составляет около 150 000тг.

Функционал вышеуказанного модуля интеграции излишне богат, что так же увеличивает стоимость самого модуля и многие функции попросту не будут задействованы в работе.

К плюсам продукта, купленного за определенную сумму денег можно

отнести только его готовность к использованию.

Минусы:

– Подавляющее большинство таких готовых изделий необходимо будет дополнительно настроить для ваших запросов, поскольку они настроены для стандартной базовой сборки 1С: Предприятие. Как показывает практика, база 1С: Предприятие значительно дорабатывается и переделывается непосредственно для заказчика, поэтому не стоит надеяться, что модуль будет работать «из коробки». Это является причиной дополнительных денежных затрат, иногда не только равных цене модуля, но и превышающие его стоимость.

– Подобные продукты изначально закрытые и чаще всего зашифрованные – нет гарантий безопасности хранимой и передаваемой информации.

– Разработчики готовых комплексов не сосредотачиваются на одном конкретном виде коммерческой деятельности, поэтому стараются учитывать множество разных аспектов работы комплекта 1С: Предприятие и интернет-магазина, что влечет за собой загромождение кода и громоздкость системы. Существует также заметная тенденция вставлять технические характеристики, изображения и другие данные в карточку продукта, которые вообще не нужно было учитывать ранее, а затем все это должно быть загружено в интернет-магазин. Это приводит к экспоненциальному увеличению передаваемых данных, что, в свою очередь, приводит к увеличению времени загрузки и выгрузки, загрузки сервера и каналов передачи данных.

Плюсы создаваемого модуля в данном дипломном проекте:

- Масштабируемость;
- Возможность быстрого внедрения;
- Отсутствие огромного количества лишних настроек
- Открытость кода

2 Технологический раздел

2.1 Архитектурный паттерн MVC-L

2.1.1 Способ разработки веб приложения

Model-View-Controller-Language (MVC-L, «Модель-Представление-Контроллер-Язык») — схема разделения данных приложения, пользовательского интерфейса и управляющей логики на 4 отдельных компонента: модель, представление, контроллер и языковую часть — таким образом, что модификация каждого компонента может осуществляться независимо (рисунок 2.1).

2.1.2 Основная структура каталогов

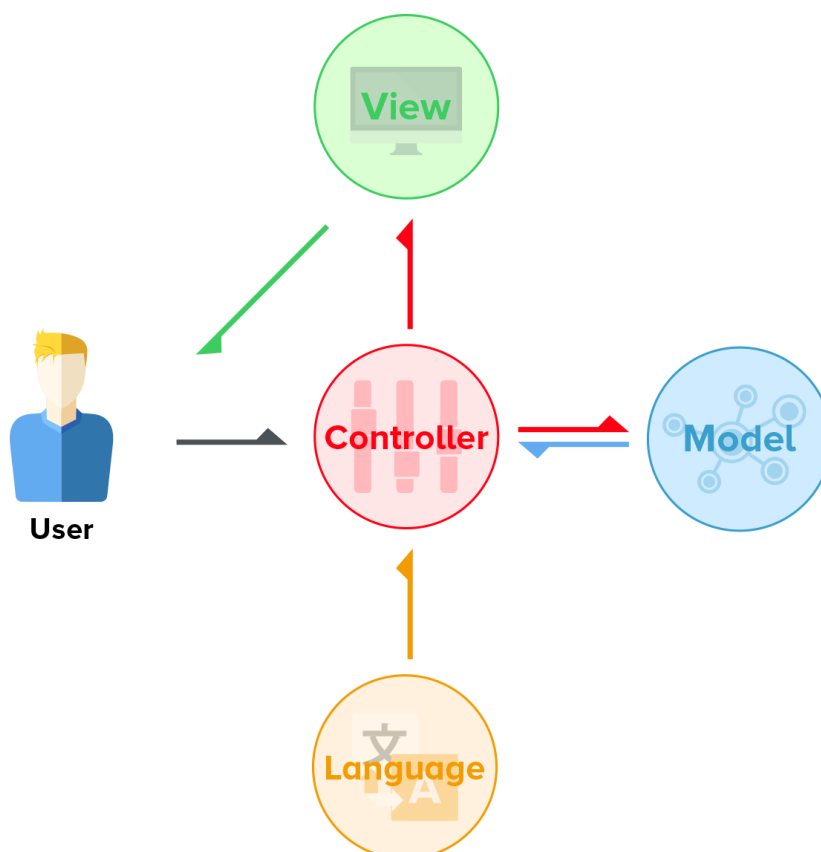


Рисунок 2.1 – Схема разделения данных приложения

Базовая структура файла для модуля будет разделена на две части: папку администратора и каталог. Содержимое каждой папки будет соответствовать структуре MVCL соответственно, с той разницей, что администратор будет заниматься исключительно функциональностью бэкэнда, а каталог - функциональностью веб-интерфейса. Пользователи модуля будут взаимодействовать и настраивать его параметры в административной части магазина. Следовательно, файлы в папке администратора будут обрабатывать любые изменения в его настройках, способ отображения модуля в администрировании, установку / удаление модуля и т.д. Аналогичным образом, способ отображения модуля, и его работа в интерфейсе пользователя. Из магазина будут обрабатываться файлы в папке каталога.

2.1.3 Controller (Контроллер)

Первый файл, который создается, будет контроллером для страницы интерфейса администратора модуля. OpenCart автоматически идентифицирует существующие модули, просто прочитав папку `admin / controller / module` магазина. Любые модули, существующие в этой папке, будут автоматически отображаться на странице «Модули» и на странице «Полномочия пользователя».

2.1.3.1 Файл контроллера

Это место, где можно загрузить языковые файлы для преобразования текста в переменные, которые будут использоваться в файле шаблона. Переменная `$_['text']` обрабатывается контроллером, а затем отправляется в виде `$text` в представление. Здесь также используется несколько файлов моделей и их функции класса, включая файл модели модуля, если он есть.

Также может быть функция, определенная как публичная функция `install ()`. Эта функция будет активирована при нажатии на ссылку установки на странице «Расширения» > «Модули». Аналогично, функция, определенная как `public function uninstall ()`, будет запущена при нажатии на ссылку удаления. Можно использовать эти функции для создания и удаления любых структур (например, таблиц базы данных или настроек конфигурации), необходимых вашему модулю. Хорошей практикой является создание функции удаления для очистки любых изменений, внесенных модулем.

2.1.3.2 Доступ через URL

Контроллер - единственный файл в структуре MVC-L, доступ к которому осуществляется через URL в OpenCart. В администрации URL будет выглядеть как `/admin/index.php?route=module/my_module&token`. Администратор добавляет токен в URL, тогда как ссылка в каталоге его не будет иметь. В результате файл контроллера будет иметь функцию, определенную как `public function index ()`. Это общедоступная «страница», загружаемая URL-адресом, которая будет отображаться при нажатии кнопки «Редактировать» и куда будет отправляться форма просмотра. Представленные данные будут обработаны в этой функции и сохранены в таблице базы данных `settings` через объект конфигурации контроллера.

2.1.4 View (Представление)

Второй необходимый файл для интерфейса администратора модуля - это файл представления. Он будет создан в папке `admin / view / template / module` и будет иметь название `exchange1c.tpl`. Это стандарт для файлов просмотра OpenCart. В этом файле создается форма для пользователя. Он будет передан в индексную функцию контроллера модуля. Самый простой способ создать файл представления - это скопировать и вставить существующий аналогичный файл представления и отредактировать форму, чтобы она содержала правильные поля для параметров конфигурации модуля.

В представлении можно получить доступ к тексту на языке, который хранится в файле контроллера в виде переменной РНР.

2.1.5 Model (Модель)

Модули редко требуют файл модели на стороне администратора. Однако, если ваш модуль использует собственную таблицу базы данных или пользовательский запрос для создания данных определенного формата, то вы можете написать файл модели. Примером этого может быть, если вы пишете модуль счетчика посетителей, где каждый посетитель хранится в таблице базы данных со своим IP-адресом и количеством посещений. Затем можно создать файл модели с функцией для определения и создания этой дополнительной таблицы в базе данных OpenCart. Файл модели будет находиться в папке `admin / model / module`.

2.1.6 Language (Язык)

OpenCart расширяет понятие MVC до MVCL, обеспечивая простой способ разделения языковой информации. Языковые файлы можно использовать для хранения любого текста, например, заголовков, названий, кнопок и т.д. Таким образом, для того, чтобы перевести магазин, нужно внести изменения в файл соответствующего языка.

2.1.7 Функциональность внешнего интерфейса

Внешний интерфейс модуля следует той же схеме, что и описанный выше интерфейс администратора. То, что будет включено в каждый из файлов внешнего интерфейса, будет в значительной степени зависеть от того, что должен делать модуль. Модуль может обращаться к любым файлам моделей, которые уже существуют в OpenCart, не нужно писать собственные запросы к базе данных, если такой же запрос уже существует. Например, каталог / модель продукта содержит много полезных запросов для получения продуктов. Использование этих функций модели должно быть предпочтительнее, чем написание новых.

Ключевое отличие во внешнем интерфейсе модуля заключается в том, что файл представления будет находиться в папке `catalog / view / template / module`.

В интерфейсной части модуля будет иметься доступ к опциям конфигурации, сохраненным модулем, через объект конфигурации контроллера и переменную `$ settings`, передаваемую в функцию индекса контроллера модуля. На основе этих настроек можно управлять аспектами отображения внешнего интерфейса внесенных модулем.

2.2 Преимущества используемых технологий

В MVC логика приложения четко отделена, что, в свою очередь, значительно упрощает работу над большими проектами.

Плюсы:

- 1) Удобство отображения разных видов (представлений) для разных типов устройств при использовании одних и тех же данных;
- 2) Облегчает поддержку и тестирование кода.

2.2.1 Преимущества PHP

Задача языка PHP - обеспечить эффективную связь веб-ресурса с сервером и базами данных. В то же время этот язык чрезвычайно прост в изучении и использовании. Фактически, все, что вам нужно знать, чтобы начать изучать php и программировать на нем, - это самый простой язык для веб-разметки HTML. PHP прекрасно сочетается с HTML-кодом. Однако выполнение скриптов одним браузером недостаточно. Требуется веб-сервер.

Код, написанный на PHP, предназначен для решения двух задач:

- 1) html-часть отвечает за появление и отображение информации;
- 2) php-часть, интегрированная в html, предоставляет возможности интерактивности и динамики.

В то же время такие программные коды и, соответственно, проекты, составленные с их помощью, являются легкими, эффективными, гибкими, многофункциональными, удобными в администрировании, редактировании и обслуживании.

Сегодня подавляющее большинство веб-сайтов, сервисов и приложений, а также популярных платформ, таких как Joomla, Drupal, OpenCart, WordPress, 1С-Bitrix и, написаны на PHP.

Язык PHP имеет несколько неоспоримых преимуществ:

- Высокая скорость и, соответственно, общая производительность ресурсов.
- Бюджет, экономия. Найти специалиста не проблема, стоимость написания программ для php невысока.
- Легко учиться, простой синтаксис.
- Отличная совместимость и переносимость - php-коды одинаково хорошо работают на разных платформах.
- Набирать текст и редактировать его можно в любом текстовом или html-редакторе.
- Высокая гибкость, вместительность и функциональность. PHP-программа может быть сделана отдельно от разработки веб-страницы, без привязки, а затем объединена. Это значительно упрощает жизнь дизайнеров и программистов.
- Многозадачность и широкие возможности - создание любых веб-приложений, блогов, гостевых книг, интернет-магазинов, сайтов, работа с перенаправлениями, заголовками, pdf-документами, базами данных, электронной почтой.

2.2.2 PHP Storm

В качестве среды разработки был выбран PHP Storm. Программное обеспечение JetBrains PhpStorm - это специализированный инструмент веб-разработки, ориентированный на веб-приложения и другие типы программ, которые можно создавать с использованием языка PHP и HTML, JavaScript и CSS. Решение PhpStorm развертывает и синхронизирует проекты по протоколу FTP. Среда PhpStorm предлагает функции для автоматического завершения языковых конструкций PHP в коде, проверки кода, различных алгоритмов рефакторинга и быстрой навигации по коду.

Графический PHP-отладчик PhpStorm поддерживает условные точки останова, отслеживание значений и автоматическую отладку отдельных процедур (рисунок 2.2).

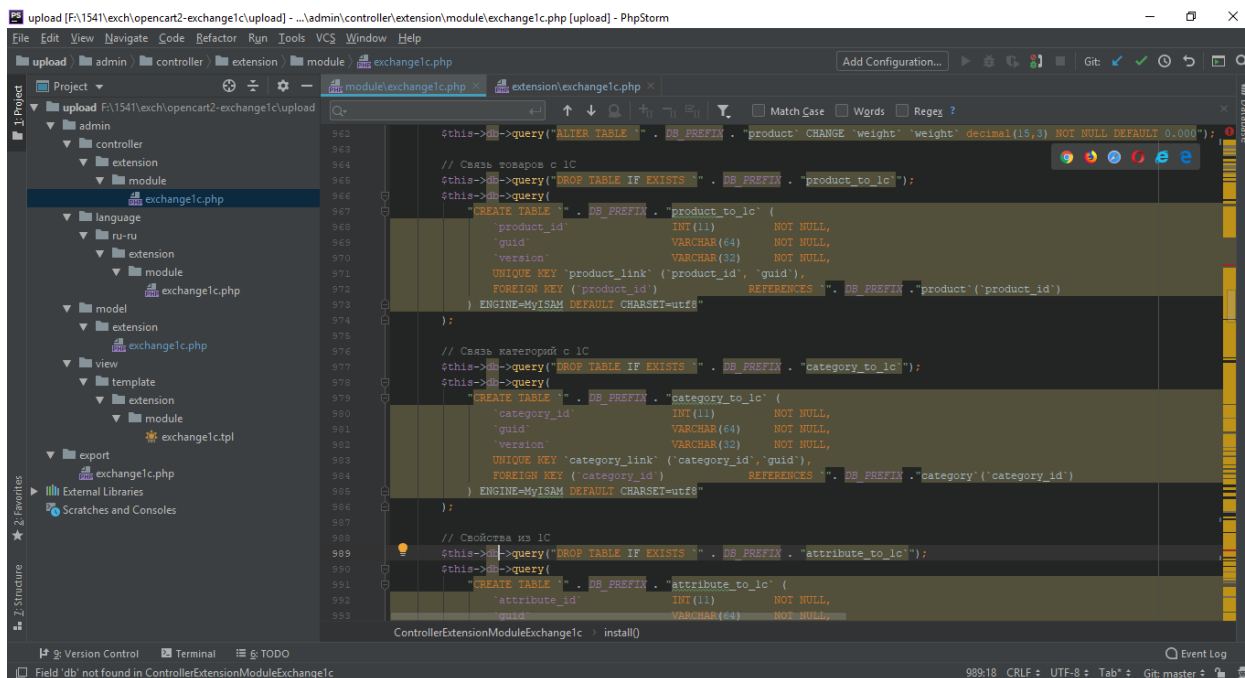


Рисунок 2.2 – Окно PHP Storm

2.3 Стандарт обмена CommerceML

Microsoft согласилась поддержать и разработать общий стандарт для обмена коммерческой информацией в формате XML между Microsoft, 1С, рядом отечественных интернет-компаний и московским офисом Intel.

Стандарты могут значительно снизить затраты на организацию информационного взаимодействия за счет унификации обмена коммерческой информацией между различными организациями: как теми, которые действуют

на рынке интернет-коммерции, так и теми, которые работают в области традиционной (оффлайн) коммерции.

Использование программного обеспечения, поддерживающего эти стандарты, торговыми организациями позволит им с минимальными усилиями и без участия программистов организовывать публикацию своих предложений на любых веб-витринах, поддерживающих этот стандарт, а также осуществлять обмен информацией между собой. без специальной ревизии программ. Например, когда товары зачисляются покупателю, информация о бизнес-операции может автоматически загружаться из данных, полученных от продавца.

Разработчики стремились максимально повысить открытость стандартов, чтобы они могли развиваться в зависимости от объективных потребностей рынка и поддерживаться максимально широким кругом производителей экономического программного обеспечения и интернет-компаний. Для этого разработчики изначально создавали стандарты независимо от собственных программных или информационных структур баз данных и исходили из общих принципов организации торговой деятельности. В то же время стандарт учитывает различные особенности работы как интернет-компаний, так и торговых организаций.

При разработке стандартов использовался ряд западных аналогов, однако предлагаемые стандарты существенно отличаются от них, поскольку они учитывают внутреннюю специфику и включают несколько универсальных решений, необходимых российским интернет-компаниям и торговым организациям. Однако стандарты CommerceML имеют много общего с решениями, используемыми в самых популярных отечественных системах электронной коммерции.

3 Проектная часть

3.1 Архитектура системы взаимодействия

Разрабатываемый модуль является элементом интернет-магазина на базе OpenCart, который разработан в использовании языка PHP версии 5.4 и в качестве БД использует MySQL версии 5.6. На рисунке 3.1 изображена архитектура системы.

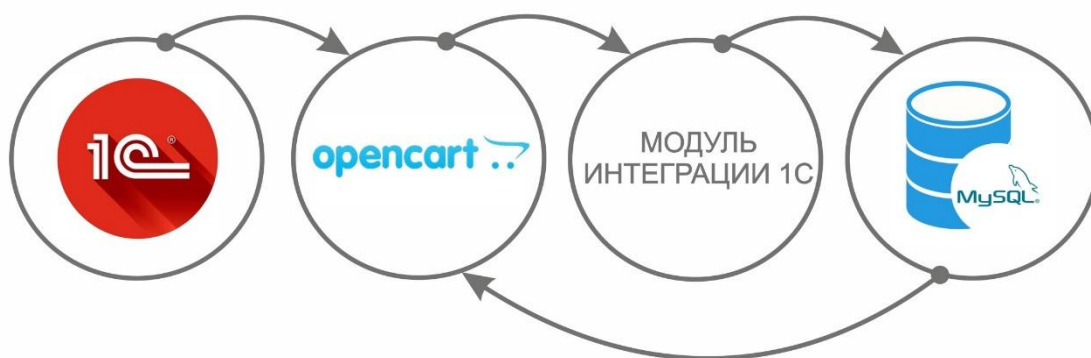


Рисунок 3.1 – Архитектура системы взаимодействия

3.2 Описание диаграммы вариантов использования

Для разработки диаграммы вариантов использования был задействован онлайн конструктор Use Case диаграмм Visual Paradigm (рисунок 3.2). В бесплатной версии конструктора есть возможность создать все виды Use Case диаграмм.

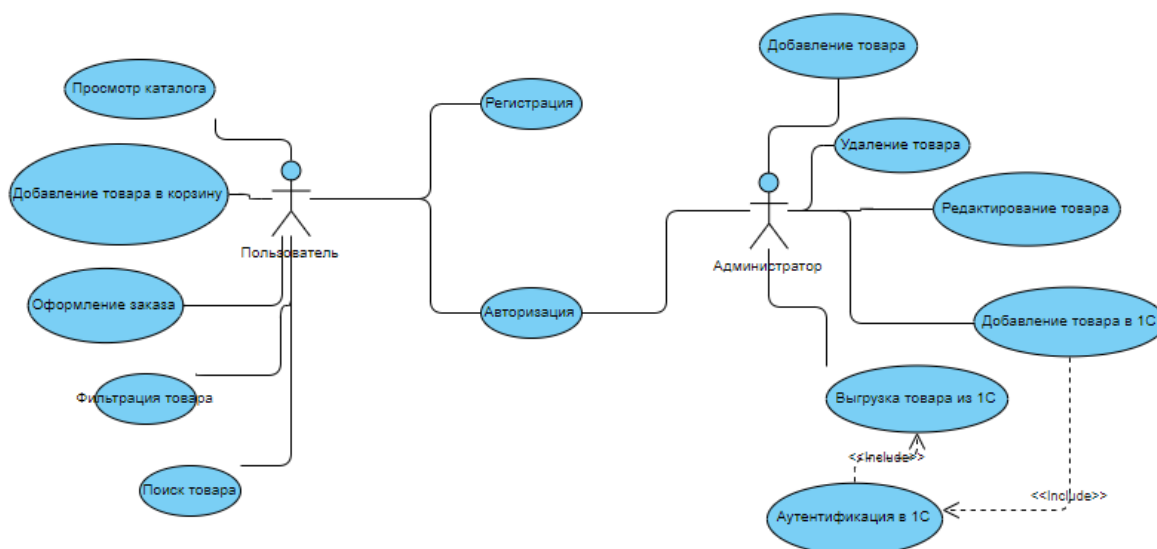


Рисунок 3.2 – Диаграмма вариантов использования

Любой пользователь, зарегистрированный или нет, свободно может просматривать товар, описание и его наличие. Как видно на рисунке 8, пользователю не обязательно регистрироваться в системе чтобы оформить заказ. Пользователь после добавления необходимого товара в корзину переходит в нее и попадает в пункт «Оформление заказа» Далее, пользователю, не желающему регистрироваться, необходимо выбрать пункт «Оформить заказ без регистрации». Это ускоряет процесс покупки. В остальном же функционал магазина схож с большинством типовых интернет-магазинов.

Администратор интернет-магазина обладает гораздо большим функционалом:

Функции, которыми может воспользоваться только авторизованный администратор:

- Добавление товара;
- Удаление товара;
- Редактирование товара.
- Выгрузка товара из 1С: Предприятие;

Именно последний пункт выделяет данный интернет-магазин среди множества других.

Данная возможность реализована с помощью разработанного модуля интеграции 1С: Предприятие. Платформа OpenCart позволяет устанавливать дополнительные модули, находящиеся в каталоге расширений, а так же модули которые каждый может разработать под собственные нужды.

Данный модуль работает непосредственно с базой данных интернет-магазина, осуществляя запись данных напрямую в таблицы БД.

ER диаграмма базы данных интернет магазина представлена на рисунке.

3.3 Разработка ER диаграммы

Для разработки ER диаграммы был использован онлайн конструктор Visual Paradigm (рисунок 3.3).

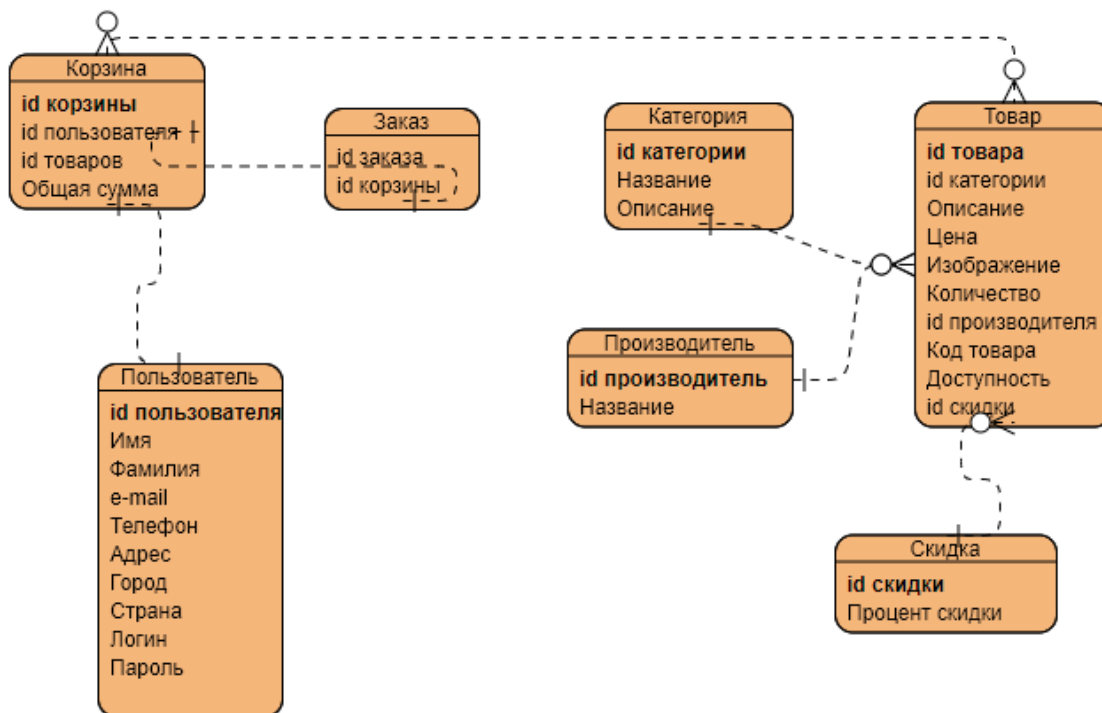


Рисунок 3.3 - ER диаграмма БД

3.4 Проектирование интерфейса

Интерфейс модуля интеграции планировалось сделать максимально простым и интуитивно понятным с описанием каждого поля и переключателя. Интуитивно непонятный и неудобный интерфейс помимо увеличения времени работы с интернет-магазином оттолкнет пользователей. При проектировании интерфейса учитывалось то, что модуль будет модернизироваться и усовершенствоваться, поэтому было заранее оставлено место под дополнительные вкладки.

Основные задачи пользовательского интерфейса:

- Обеспечить максимальную информативность;
- Сократить время на работу с модулем за счет простоты настроек;

Проектирование интерфейса модуля можно разделить на следующие этапы:

- Создание концепции;

– Создание прототипа.

3.4.1 Создание концепции

На этом шаге определяются основные элементы шрифты, цвета, отступы, расположение полей. Концепция интерфейса основывалась на главных чертах раздела администрирования, были использованы подобные цветовые решения и шрифты текста (рисунок 3.4).

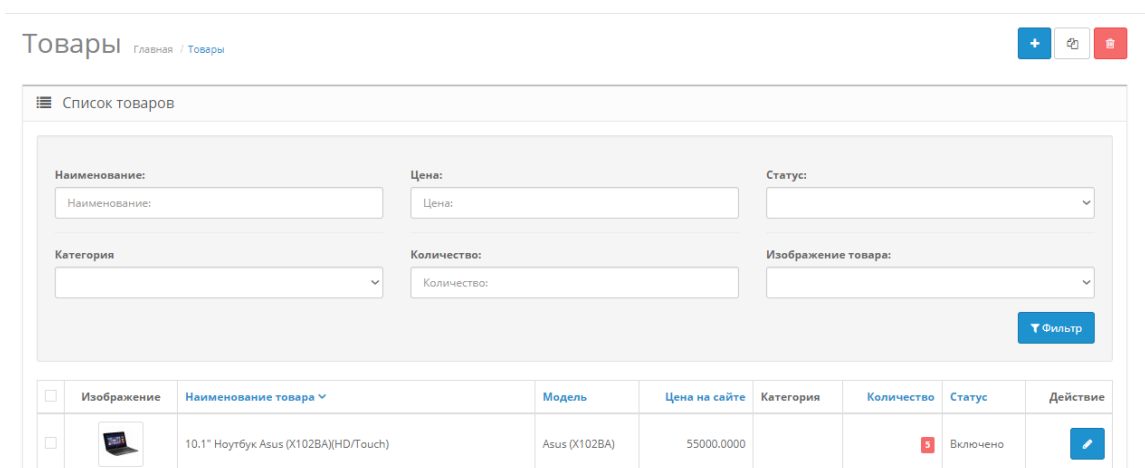


Рисунок 3.4 – Пример частых ошибок приложения

3.4.2 Создание прототипа

Прототипирование интерфейса велось с помощью Moqups (рисунок 3.5). Moqups – это оптимизированное веб-приложение, которое помогает создавать макеты, макеты, диаграммы и прототипы

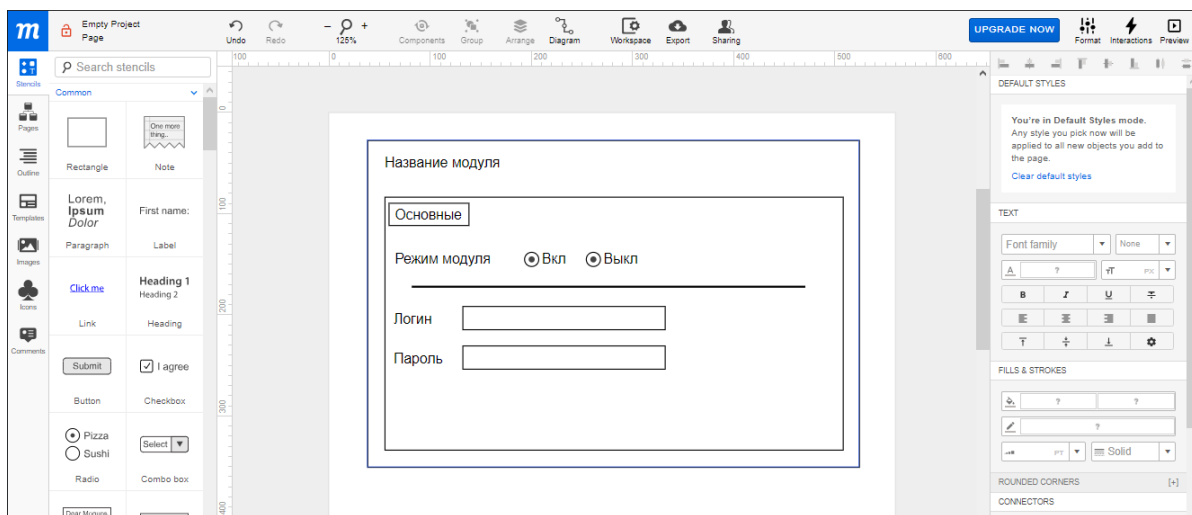


Рисунок 3.5 – Создание прототипа в Moqups

Интерфейс модуля разрабатывался в соответствии с прототипом созданным ранее. Главная страница модуля содержит в себе переключатель (Radio Button) состояния, который отвечает за активность модуля, а также поля для авторизации при подключении из УС 1С: Предприятие (рисунок 3.6).

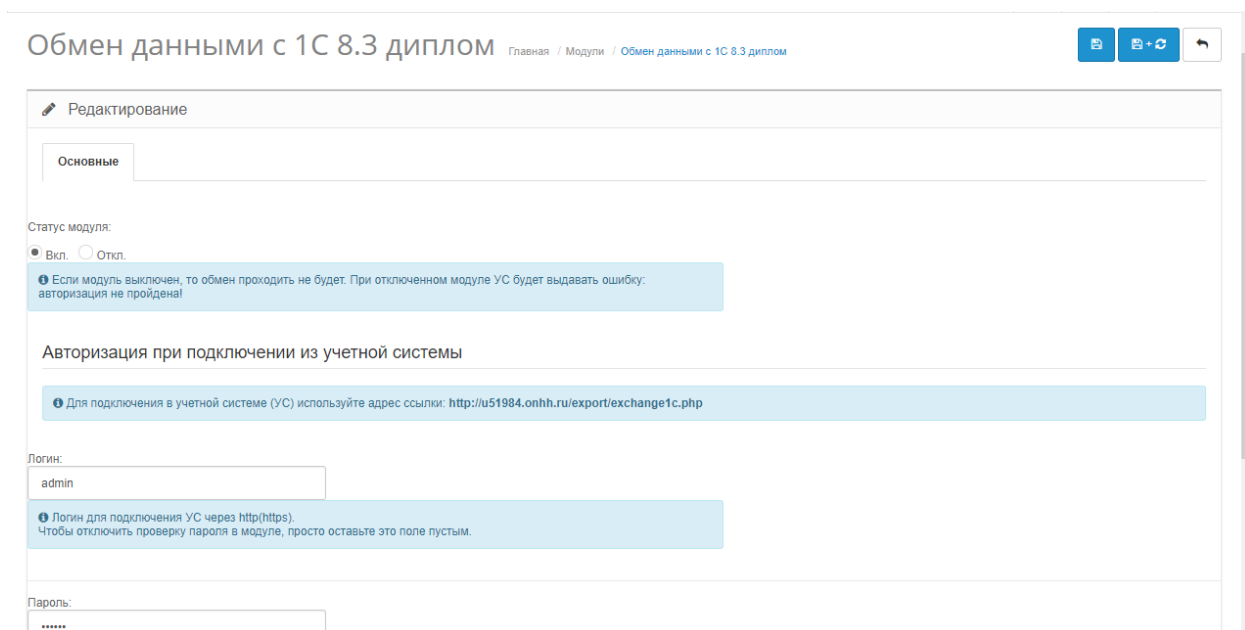


Рисунок 3.6 – Прототип главной страницы модуля

4 Экспериментальный раздел

4.1 Функции модуля интеграции 1С: Предприятие

Изначальной целью дипломного проекта было создание модуля и обеспечение его функционалом достаточным для автоматизации выгрузки номенклатуры из УС 1С: Предприятие.

План работы:

- сделать программу выгрузки из 1С: Предприятие только тех данных, которые будут учитываться в магазине;
- загрузить эти данные в базу интернет магазина;
- автоматизировать процесс загрузки и выгрузки.

Конкретизируем каждый пункт и подробно распишем его реализацию.

4.2 Выгрузка из 1С: Предприятие

Задача загрузки состоит из следующих шагов:

- Загрузка в файл списка данных, содержащего код продукта, название, артикул, производителя, различные типы цен (партнерская цена, оптовая, мелкооптовая и розничная торговля), количество, релевантность продукта - независимо от того, доступен он или нет (вы также можете добавить позже товары в пути или предварительный заказ) (рисунок 4.1).
- Отправка файла в папку интернет-магазина.

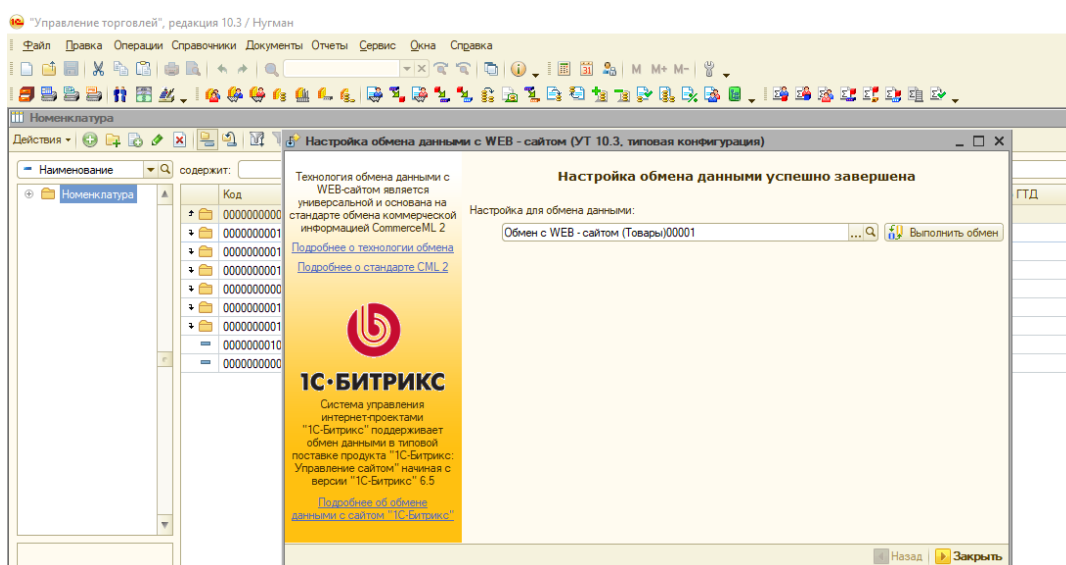


Рисунок 4.1 – Окно настройки обмена данными 1С: Предприятие

Бывает, что категории товаров в базе данных 1С: Предприятие выглядят по одному, и в магазине необходимо, чтобы категории были представлены более подробно, или наоборот. Например, мыши, клавиатуры, наборы мышь + клавиатура называются манипуляторами, и в интернет-магазине все это должно быть отдельным, как и во многих других продуктах. Чтобы не нарушать уже сложившийся способ работы, мы должны придумать обходные маневры, поэтому для дальнейшего удобства мы немного отклонимся от стандартной задачи - категории товаров в 1С: Предприятие будут одной, а в интернет-магазине они будут быть совершенно другим. Для этого в карточке товара создается поле «Категория Интернета», в котором из каталога будут заполнены категории товаров для интернет-магазина. Бывают случаи, когда в базе данных накапливается много товаров за несколько лет, и большая часть этого товара просто не используется и никогда не появится. Таким образом, интернет-категория прикрепляется только к продукту, который доступен, и продукт будет выгружен, в котором есть интернет-категория. За счет этого количество загружаемой информации будет сокращено.

Можно загружать данные в формате XML. Это простой текстовый файл, в котором ячейки таблицы записываются через разделитель, поэтому он мало весит и, что очень удобно, в 1С: Предприятие есть функции для работы с файлами XML. Обычно точка с запятой используется в качестве разделителя, но иногда возникает проблема - точка с запятой может появиться в описании продукта, например:

MSI M34-RS5 SDD (MS5,GTX1060, Force630a, video, 4*DDR4;

PCIx16; 4*SATAIII; RAIDx0;Lan;bch;mATX), что в последствии, приведет к неправильной загрузке данных и захламлению хранилища информационным мусором из-за сдвига ячеек. В Интернете есть решение этой проблемы в виде написания кучи проверок, что приводит к неоправданному росту программного кода. Лучше пойти другим путем, исправить проблему - поставить другой разделитель. Выбор пал на такой разделитель ^. Этот символ, так называемый «дом», в названии продукта и в других данных, которые будут выгружены, не встречается вообще. Также можно заменить разделитель несколькими символами, идущими подряд, тогда вероятность их появления будет очень и очень мала, например - ~ ^ &.

4.3 Загрузка данных из созданного 1С: Предприятие файла в базу интернет магазина

Необходимо создать скрипт на php, который будет запускаться в папке и работать с базой интернет-магазина.

Скрипт загружает файл сформированный 1С: Предприятие считывает его построчно, проверяет наличие прочитанного товара и записывает/создает товар в базе данных магазина. Для этого сначала проверяется наличие загруженного

файла, если таковой имеется, загружается в двумерный массив, и данные товаров записываются в базу данных магазина с использованием SQL-запросов.

В магазине есть много таблиц, которые необходимо заполнить или переписать. Они абсолютно не такие как в 1С: Предприятие. Например, есть отдельная таблица производителей, ссылки производителей и продуктов и так далее. Поэтому практически все данные, полученные из файла 1С: Предприятие, должны быть не только помещены в необходимые таблицы, но и учитывать связи между этими данными. Для этого нужно будет узнать коды категорий товаров, производителя и т.д., которые сам магазин устанавливает автоматически при заполнении соответствующих таблиц.

После того, как скрипт заполняет двумерный массив, он подключается к базе данных хранилища. Затем массив читается построчно, где каждая строка является отдельным элементом. Чтобы избежать заполнения дубликатами товара, необходимо проверить наличие товара. Удобнее сначала проверить, есть ли такой производитель в таблице производителей. Если нет производителя, очевидно, нет продукта. Если производителя нет, в таблице производителей создается новая запись. Аналогичным образом, в будущем вам придется создать новый продукт в таблице продуктов, цены в таблице цен для этого продукта и назначить категорию в таблице категорий.

Если имеются загруженные продукты, то полученные данные по этому продукту просто перезаписываются согласно соответствующим таблицам БД. Затем происходит переход к следующей строке и повторяется весь способ проверки, записи или перезаписи данных в таблице. И так до конца файла.

В результате появился интернет-магазин с определенным набором товаров и технических спецификаций (если они нужны) и картинок.

4.4 Автоматизация

После написания и проверки программы этот процесс будет автоматизирован: по истечении определенного периода времени файл данных будет выгружен из 1С: Предприятие и автоматически загружен в БД, данная возможность реализуется за счет встроенной функции 1С: Предприятие «Периодический обмен данными» (рисунок 4.2). Обратного процесса передачи данных из магазина в 1С: Предприятие не произойдет, потому что когда вы покупаете товар, уведомление приходит по почте, а затем менеджер берет на себя обязанности по обработке заказа с покупателем и создать документы нужные для работы.

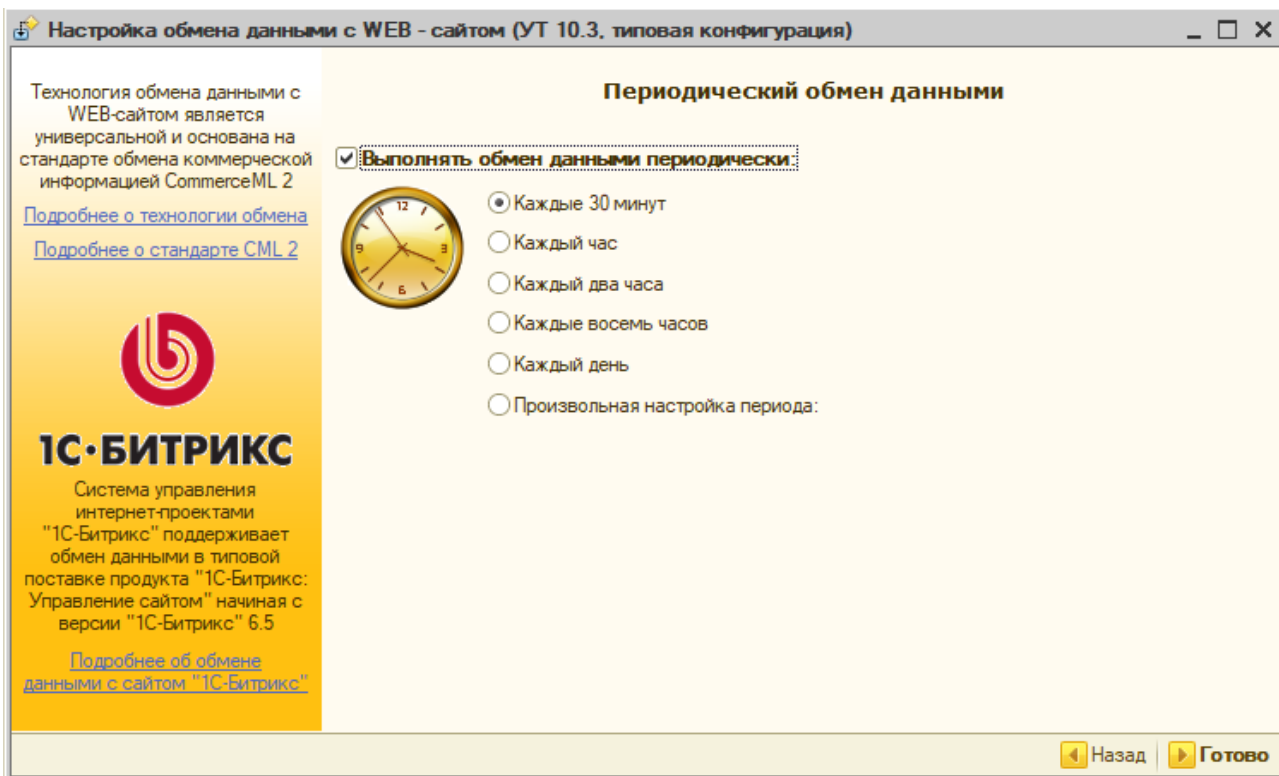


Рисунок 4.2 – Окно периодического обмена данными

4.5 Алгоритм выгрузки данных на сайт

Процесс выгрузки данных из учетной системы на сайт можно представить в виде следующей схемы (рисунок 4.3):

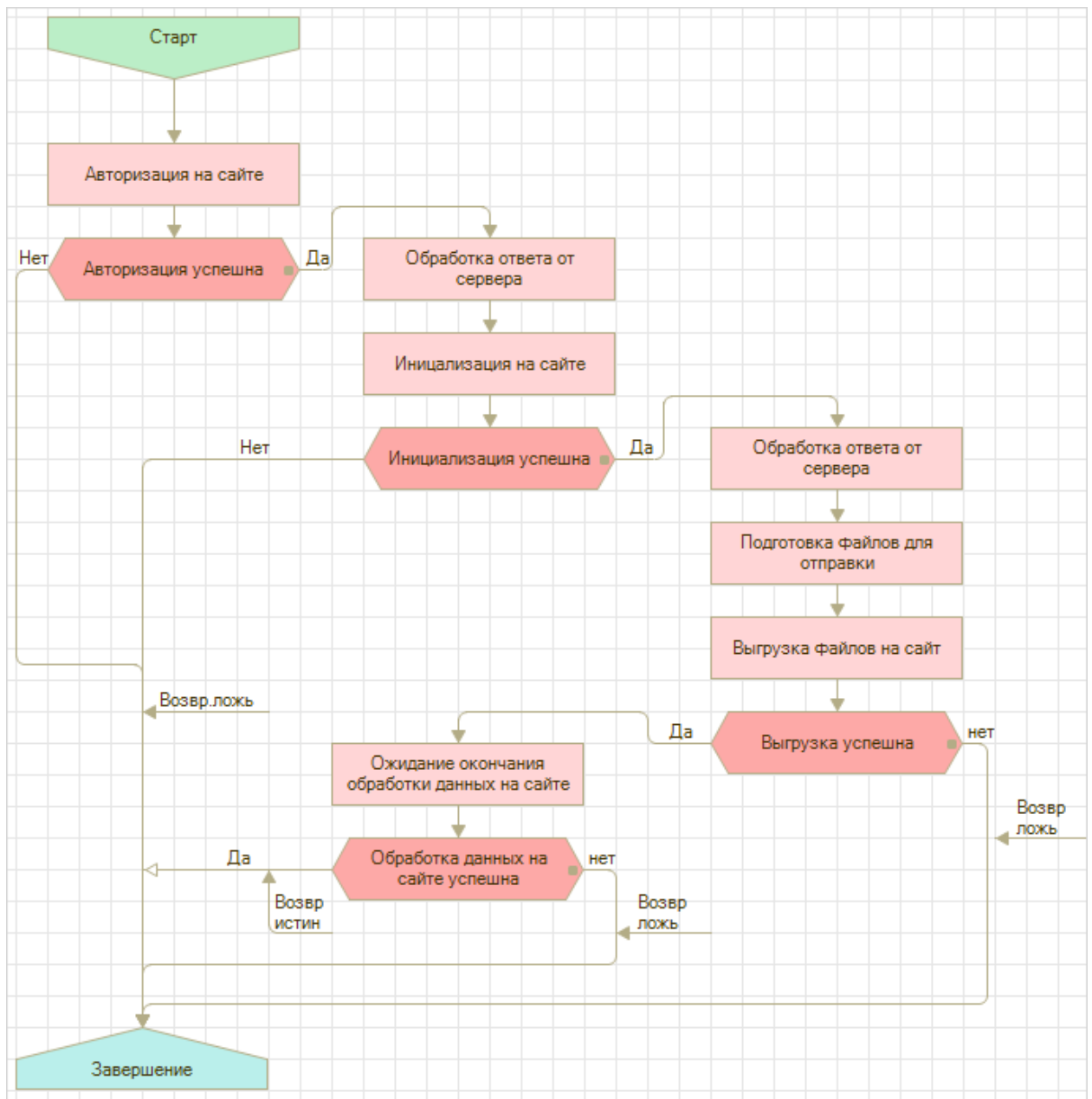


Рисунок 4.3 – Алгоритм выгрузки данных на сайт

4.5.1 Авторизация на сайте

На этапе Авторизация на сайте УС получает ресурс из указанного адреса. Адрес генерируется по такому формату (рисунок 4.5): <Адрес_скрипта> + "?type=" + <Тип_соединения> + "&mode=checkauth", где:

<Тип_соединения> может принимать значения catalog или sale в зависимости от того, что нужно выгружать (sale используется для выгрузки документов);

<Адрес_скрипта> - адрес, указанный в настройке обмена.

При успешной авторизации (рисунок4.4) сайт возвращает временный файл с данными:

- вторая строка содержит имя cookie файла;
- третья строка содержит значение cookie файла;
- четвертая строка содержит ключ сессии обмена (CSRF);
- пятая строка содержит дату и время сервера сайта (CSRF).

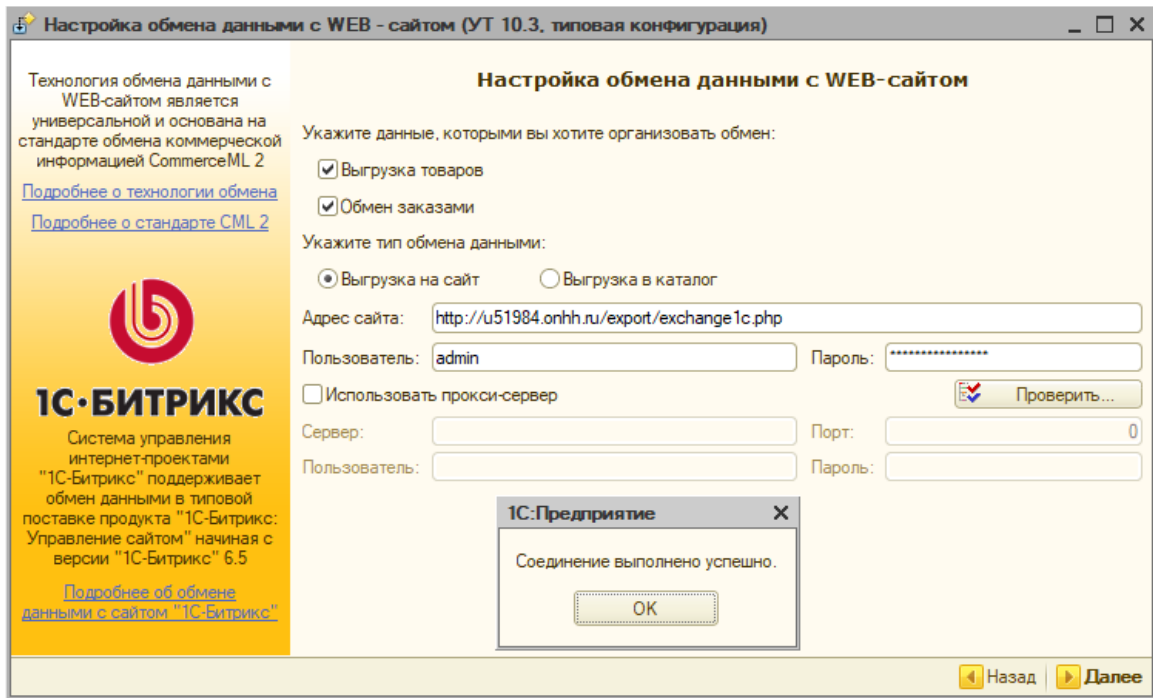


Рисунок 4.4 – Страница настройки обмена данными

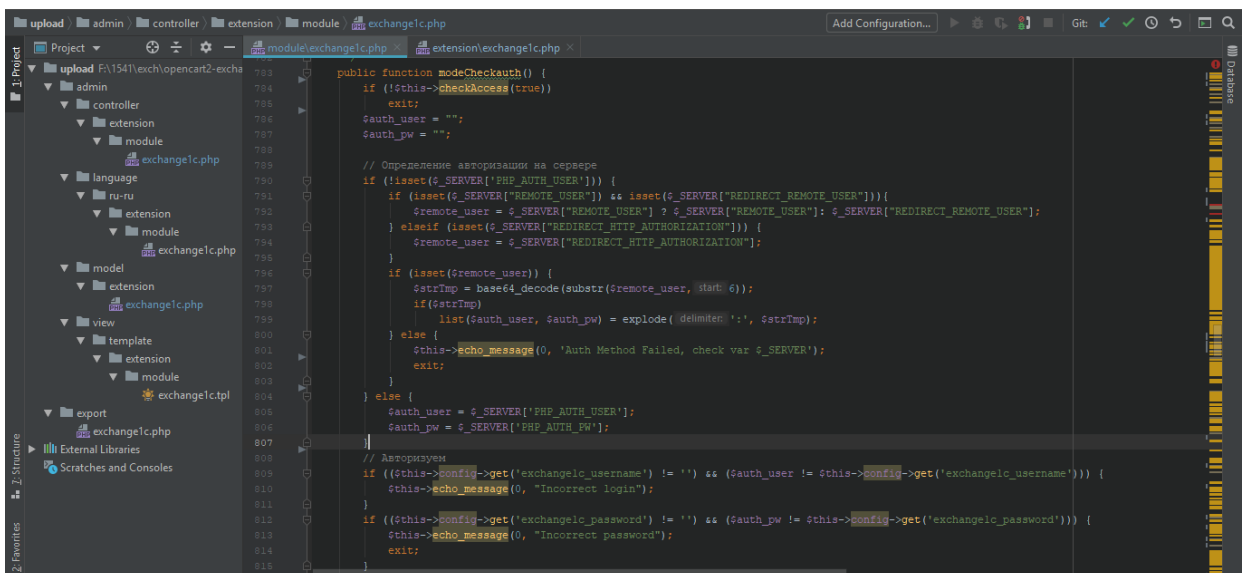


Рисунок 4.5 – Фрагмент авторизации на сайте

4.5.2 Инициализация на сайте

На этапе Инициализация на сайте учетная система получает ресурс из указанного адреса, а также сообщает о версии CommerceML. Адрес генерируется по следующему формату: <Адрес_скрипта> + "?type=" + <Тип_соединения> + "&mode=init" + "&" + <Ключ_сессии> + "&version=" + <Версия_CommerceML>, где:

<Тип_соединения> может принимать значения catalog или sale в зависимости от того, что нужно выгружать;

<Адрес_скрипта> - адрес, указанный в настройке обмена;

<Ключ_сессии> - ключ сессии, полученный на этапе Авторизация на сайте;

<Версия_CommerceML> - версия структуры XML-файлов обмена.

Текущая версия - 3.1. Если не указать версию, то будут возвращены файлы версии 2.1.

Заголовок запроса (рисунок 4.6) формируется по следующему алгоритму: "Cookie: " + CookieName + "=" + CookieValue, где вся информация о cookie берется с этапа Авторизация на сайте.

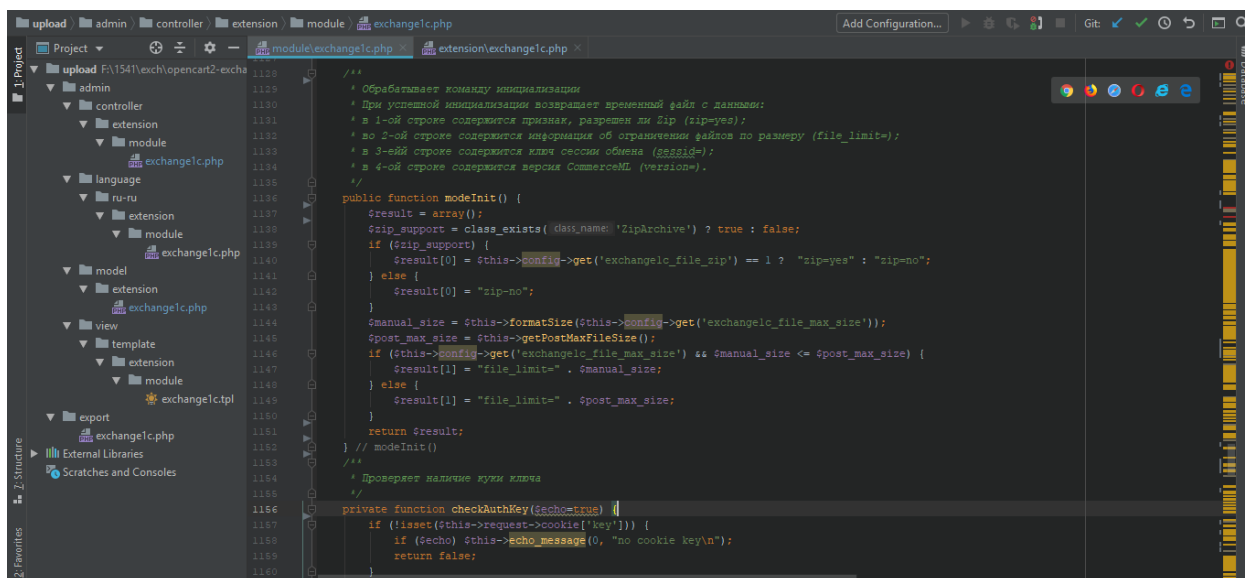
При успешной инициализации возвращает временный файл с данными:

первая строка содержит свойство, разрешен ли Zip (zip=yes);

вторая строка содержит информацию об ограничении файлов по размеру (file_limit=);

четвертая строка содержит в себе ключ сессии обмена (sessid=);

пятая строка содержит в себе версию CommerceML (version=).



```
1128 // Обрабатывает команду инициализации
1129 // При успешной инициализации возвращает временный файл с данными:
1130 // в 1-ой строке содержится признак, разрешен ли Zip (zip=yes);
1131 // во 2-ой строке содержится информация об ограничении файлов по размеру (file_limit=);
1132 // в 3-ей строке содержится ключ сессии обмена (sessid=);
1133 // в 4-ой строке содержится версия CommerceML (version=).
1134 //
1135
1136 public function modeInit() {
1137     $result = array();
1138     $zip_support = class_exists('ZipArchive') ? true : false;
1139     if ($zip_support) {
1140         $result[0] = $this->getConfig('exchange1c_file_zip') == 1 ? "zip=yes" : "zip=no";
1141     } else {
1142         $result[0] = "zip=no";
1143     }
1144     $manual_size = $this->formatSize($this->getConfig('exchange1c_file_max_size'));
1145     $post_max_size = $this->getPostMaxFileSize();
1146     if ($this->getConfig('exchange1c_file_max_size') && $manual_size <= $post_max_size) {
1147         $result[1] = "file_limit=" . $manual_size;
1148     } else {
1149         $result[1] = "file_limit=" . $post_max_size;
1150     }
1151     return $result;
1152 } // modeInit()
1153 //
1154 // Проверяет наличие куки ключа
1155 //
1156 private function checkAuthKey($echo=true) {
1157     if (!isset($this->request->cookie['key'])) {
1158         if ($echo) $this->echo_message(0, "no cookie key\n");
1159         return false;
1160     }
1161 }
```

Рисунок 4.6 – Фрагмент инициализации на сайте

4.5.3 Выгрузка файлов на сайт

На этапе Выгрузка файлов на сайт учетная система отсылает сформированные и разбитые файлы XML в заданный адрес. Адрес генерируется по следующему формату: <Адрес_скрипта> + "?type=" + <Тип_соединения> + "&mode=file&filename=" + <имя_файла> + "&" + <Ключ_сессии>, где:

<Тип_соединения> может принимать значения catalog или sale в зависимости от того, что нужно выгружать;

<Адрес_скрипта> - адрес, указанный в настройке обмена;

<Ключ_сессии> - ключ сессии, полученный на этапе Авторизация на сайте.

Заголовок запроса формируется по следующему алгоритму: "Cookie: " + КукиИмя + "=" + КукиЗначение, где вся информация о куки берется с этапа Авторизация на сайте.

Сайт может вернуть ответ:

success - файл доставлен;

failure - файл не доставлен;

если ничего не пришло, то файл также не доставлен.

4.5.4 Ожидание окончания обработки данных на сайте

На этапе Ожидание окончания обработки данных на сайте учетная система проверяет обработал ли сайт данные (рисунок 4.7). Отслеживается каждый выгруженный файл из учетной системы до тех пор, пока не будет получен ответ об успешном завершении обработки файла, неудачном завершении или ответ не будет получен вовсе. Адрес генерируется по следующему формату: <Адрес_скрипта> + "?type=" + <Тип_соединения> + "&mode=import&filename=" + <имя_файла> + "&" + <Ключ_сессии>, где:

<Тип_соединения> может принимать значения catalog или sale в зависимости от того, что нужно выгружать;

<Адрес_скрипта> - адрес, указанный в настройке обмена;

<Ключ_сессии> - ключ сессии, полученный на этапе Авторизация на сайте.

Сайт может вернуть ответ, в котором в первой строке может содержаться: progress - файл обрабатывается (на следующей строке ответа указано на каком этапе);

success - файл успешно обработан;

failure - файл не обработан;

если ничего не пришло, то файл также не обработан.

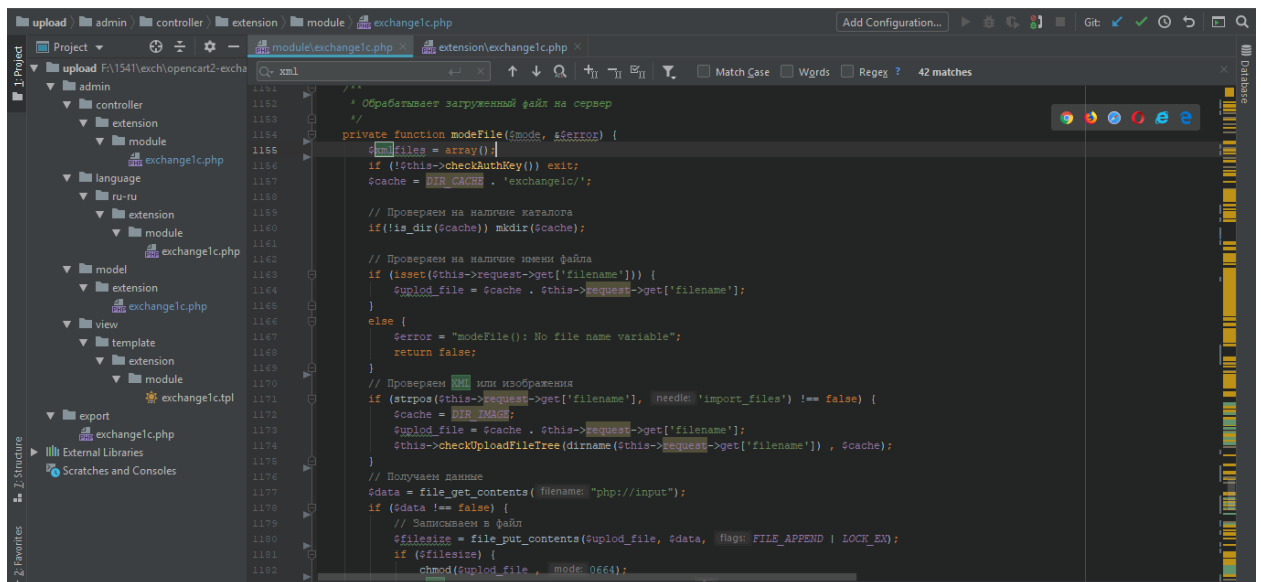


Рисунок 4.7 – Фрагмент обработки загруженного файла

Если происходит полная выгрузка, то после успешной выгрузки информации о товарах, происходит деактивация незагруженных товаров и всей информации по ним. Деактивируются все элементы, дата модификации которых меньше даты выгрузки. Дата выгрузки берется из первой авторизации на сайте (первого пакета информации о товарах): <Адрес_скрипта> + ?type=catalog&mode=deactivate + "xamp="" + <Дата_и_время_сервера_в_формате_timestamp> + "&" + <Ключ_сессии>, где:

- <Адрес_скрипта> - адрес, указанный в настройке обмена;
- <Ключ_сессии> - ключ сессии, полученный на этапе Авторизация на сайте;
- <Дата_и_время_сервера> - дата и время сервера, когда была произведена первая авторизация. Все данные о товарах, дата модификации которых меньше этого параметра, будут деактивированы.

Затем в работу включаются SQL-запросы к различным таблицам и вносятся информация о товаре. (рисунок 4.8)

```

716
717
718 // Добавляет товар в базу
719 // Возвращает product_id
720
721 private function addProduct($data) {
722     $data['status'] = $this->config->get('exchange1c_product_new_status_disable') ? 0 : 1;
723     // Подготавливаем список полей по которым есть данные
724     $fields = $this->prepareQueryProduct($data);
725     if ($fields) {
726         if (isset($data['product_id'])) {
727             $fields = "product_id = " . $data['product_id'] . (empty($fields) ? "" : ", " . $fields);
728         }
729         $this->query("INSERT INTO " . DB_PREFIX . "product SET " . $fields . ", 'date_added' = " . $this->NOW . ", 'date_modified' = " . $this->NOW);
730         $product_id = $this->db->getLastId();
731     } else {
732         // Если нет данных
733         return 0;
734     }
735     // ПРОИЗВОДИТЕЛЬ
736     if (isset($data['manufacturer_name'])) {
737         $data['manufacturer_id'] = $this->setManufacturer(htmlspecialchars($data['manufacturer_name']));
738     }
739     // Сформируем SEO
740     if ($this->config->get('exchange1c_seo_product_mode') != 'disable') {
741         $update_fields = $this->seoGenerateProduct($product_id, $data);
742         if ($update_fields) {
743             $this->compareArraysData($data, $update_fields);
744         }
745     }
746     // Связь с 1С только по Id объекта из торровой системы
747     $sql = "INSERT INTO " . DB_PREFIX . "product_to_1c SET 'product_id' = " . (int)$product_id . ", 'guid' = " . $this->db->escape($data['guid']);
748     if (isset($data['version'])) {
749         $sql .= ", 'version' = " . $this->db->escape($data['version']) . " ";
750     }
751 }

```

Рисунок 4.8 – Фрагмент добавления товара в базу

4.5.5 Распаковка изображений

Изображения помещаются в zip архив при создании файла импорта 1С: Предприятие.

При распаковке файлов на сайте проверяется является ли данный файл php скриптом или картинкой и после этого происходит присвоение данной картинке в качестве опции к товару по его id (рисунок 4.9).

```

775
776
777 // **
778 // Распаковываем картинки
779
780 private function extractImage($zipArc, $zipEntry, $name) {
781     $error = "";
782     if (substr($name, -1) == "/" ) {
783         // проверим каталог
784         if (is_dir($filename = DIR_IMAGE . $name)) {
785             // каталог существует
786         } else {
787             $mkdir = mkdir($filename = DIR_IMAGE . $name, 0775) or die ($error = "Ошибка создания директории '" . DIR_IMAGE . $name . "'");
788             if ($error) return $error;
789         }
790     } elseif ($zipEntry->open($zipArc, $zipEntry, 'r')) {
791         $dump = $zipEntry->read($zipEntry, $zipEntry->filesize($zipEntry));
792         $error = $this->checkDirectories($name);
793         if ($error) return $error;
794         if (is_file($filename = DIR_IMAGE . $name)) {
795             // Если файл существует
796             $size_dump = strlen($dump);
797             $size_file = filesize($filename = DIR_IMAGE . $name);
798             if ($size_dump != $size_file) {
799                 // файл был изменен, нужно заменить
800                 $fd = fopen($filename = DIR_IMAGE . $name, 'wb');
801                 if ($fd === false) {
802                     return "Ошибка записи файла: '" . DIR_IMAGE . $name . "'";
803                 }
804                 fwrite($fd, $dump);
805                 fclose($fd);
806             }
807             if ($this->config->get('exchange1c_log_level') == 3) {
808                 // проверим записанный размер файла
809                 $new_size_file = filesize($filename = DIR_IMAGE . $name);
810             }
811         }
812     }
813 }

```

Рисунок 4.9 – Фрагмент выгрузки изображений

4.6 Добавление номенклатуры в 1С: Предприятие

Для того чтобы добавить товар в номенклатуру, необходимо выбрать нужную группу номенклатуры справа в списке и в центральном окне нажать на кнопку «Создать». После этого появится окно «Создание номенклатуры».

Необходимо заполнить следующие пункты:

- Группа номенклатуры — будет выбрана автоматически, в зависимости от того, какая группа выбрана в списке групп слева. Также Вы можете изменить группу номенклатуры, в которую будет входить данный товар, нажав на кнопку с изображением троеочия;
- Вид номенклатуры — будет отмечен автоматически;
- Наименование новой номенклатуры;
- Единица хранения — выберите необходимую единицу хранения, в соответствии с характеристиками товара (шт., кг, м и т.д.);
- Свойства товара;
- Описание;
- Цена;

После заполнения необходимых полей только что созданной номенклатуры нужно нажать на кнопку «Записать», а затем кнопку «Ок» (рисунок 4.10)

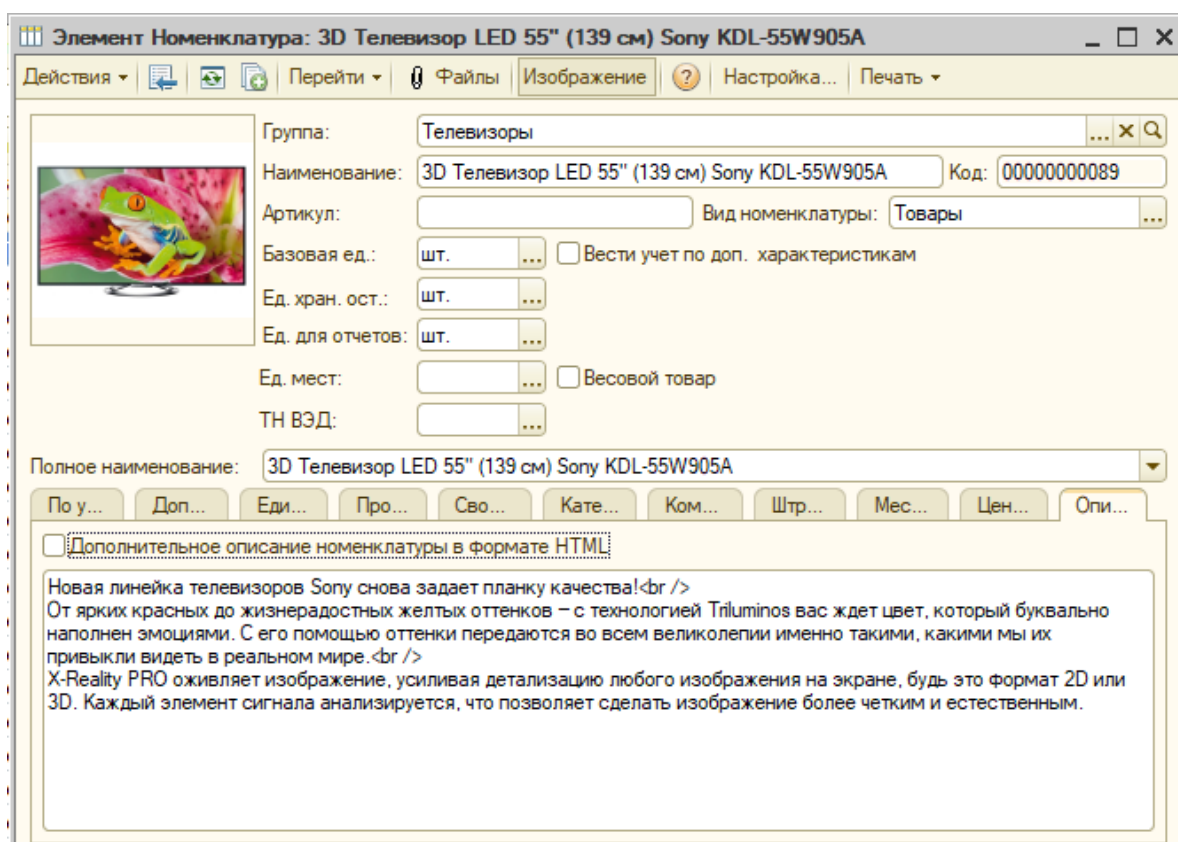


Рисунок 4.10 – Создание номенклатуры

4.7 Тестирование работы модуля интеграции 1С: Предприятие

Результат работы данного модуля показан на рисунках 4.11, 4.12, 4.13.

В результате выгрузки номенклатуры из УС в интернет-магазин были добавлены новые категории и подкатегории товаров, созданы новые позиции с названием, описанием, изображением и характеристиками.

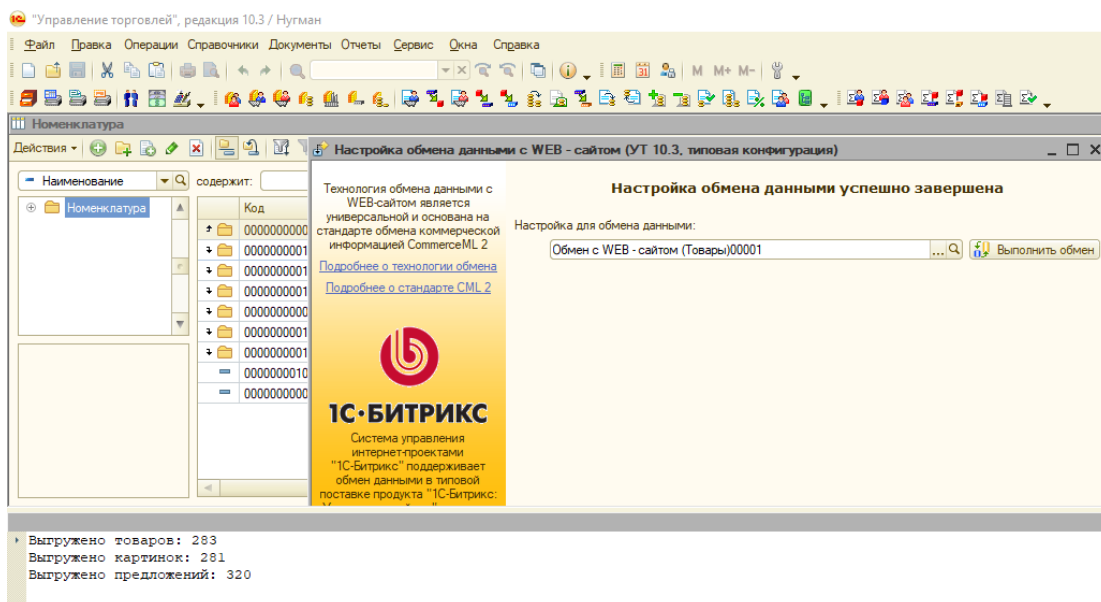


Рисунок 4.11 – Этап отправки файла импорта

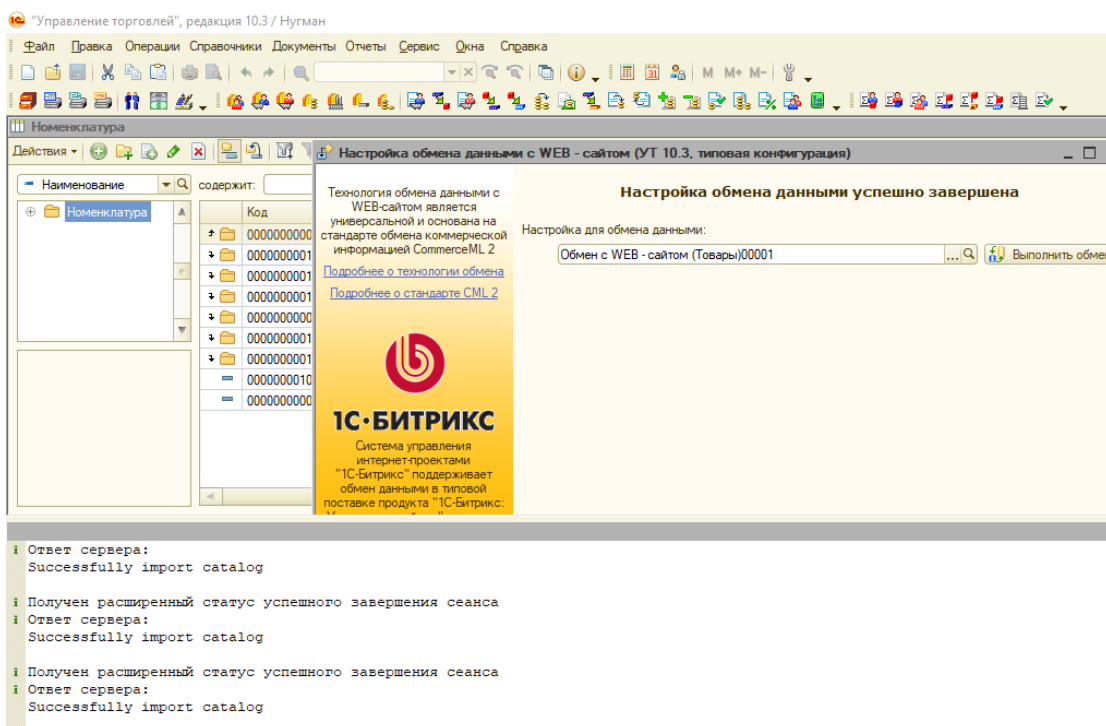



Рисунок 4.12 – Этап выгрузки изображений

Товары (283)
- Автотовары (20)
- Видеокамеры (20)
- Кондиционеры (12)
- Ноутбуки, компьютеры и программное обеспечение (136)
- Телевизоры (29)
- Фотоаппараты, объективы, аксессуары (64)
Ноутбуки (0)
Компоненты (0)
Планшеты (0)
Программное обеспечение (0)
Телефоны и PDA (0)
Камеры (0)

Сравнение товаров (0)
Сортировать: По умолчанию
Показывать: 15




3D Телевизор LED 55" (139 см) Sony KDL-55W905A

Новая линейка телевизоров Sony снова задает планку качества! От ярких красных до жизнерадостных желтых..

Отг.

[В КОРЗИНУ](#) [❤](#) [≡](#)




Телевизор LED 19" (48 см) Samsung UE19F4000

Оцените кристально-чистое изображение на экране вашего ТВ с помощью функции Samsung Clean View. Техн..

Отг.

[В КОРЗИНУ](#) [❤](#) [≡](#)



Телевизор LED 19" (48 см) Samsung UE19H4000

Фильтрация шумов для комфортного просмотра. Оцените кристально-чистое изображение на экране Вашего ТВ..

Отг.

[В КОРЗИНУ](#) [❤](#) [≡](#)

Рисунок 4.13 – Результат работы модуля

ЗАКЛЮЧЕНИЕ

В рамках данного дипломного проекта поставленная цель выполнена успешно, был разработан и внедрен модуль обмена данными с УС в структуру интернет-магазина на базе OpenCart.

Было определено место электронных магазинов в системе электронной коммерции, изучен вопрос практической разработки электронных магазинов.

Результатом дипломного проекта является спроектированный интернет-магазин, который позволяет пользователям совершать покупки онлайн. Интернет-магазин будет интегрирован с рабочей конфигурацией 1С: Предприятие: Управление торговлей 10.3.

Реализован веб-сайт с использованием выбранной системы управления контентом. Внедрены средства интеграции данных с веб-сайтом для обмена информацией о заказах и номенклатуре с конфигурацией на платформе 1С: Предприятие;

В будущем планируется расширение функционала данного модуля за счет добавления двухстороннего обмена номенклатурой, на тот случай если компания уже имеет собственный интернет-магазин, но не имеет УС. В таком случае будет возможно наполнение базы УС уже имеющимися товарами из интернет-магазина.

Помимо этого, был проведен краткий анализ предприятия, анализ интерактивного бизнеса, его свойств и видов. Были рассмотрены различные виды SMS, найдены достоинства и недостатки, рассмотрены различные способы интеграции "1С: Предприятие: Предприятие" и интернет-магазина и выявлены наиболее значимые аспекты интеграции.

Данный модуль предназначен в первую очередь для сокращения времени работы с интернет-магазином, за счет автоматизированной загрузки полной иерархии категорий, изображений, описания товара и его цены. Так же дополнительным преимуществом является то, что в УС 1С: Предприятие: Предприятие не нужно устанавливать дополнительных расширений или вносить какие-либо изменения в ее конфигурацию, что в свою очередь избавляет компанию от риска лишиться поддержки со стороны компании 1С: Предприятие.

При разработке данного модуля использовался язык программирования PHP, который интенсивно применяется для разработки веб-приложений.

Для создания интернет-магазина использовалась CMS «OpenCart», имеющая широчайший функционал, что позволило быстро и эффективно выполнить все поставленные задачи.

СПИСОК ИСПОЛЬЗОВАННОЙ ЛИТЕРАТУРЫ

- 1 Радченко М.Г. «1С: Предприятие: Программирование для начинающих: Детям и родителям, менеджерам и руководителям – Разработка в системе «1С: Предприятие:Предприятие 8.3»» 1-е изд. (2017). –Петербург: Изд-во «1С: Предприятие–Паблишинг», 2017. – 220 с
- 2 Р. Никсон «Создаем динамические веб-сайты с помощью PHP, MySQL, Javascript и CSS» 3-е изд. –Москва: Изд-во Вильямс, 2016. – 848 с.
- 3 М.Г. Радченко, Е.Ю. Хрусталева 1С: Предприятие: Предприятие 8.3. Практическое пособие разработчика. Примеры и типовые приемы. М.: ООО «1С: Предприятие-Паблишинг», 2013. – 943 с.
- 4 Стив Суэринг, Тим Конверс, Джойс Парк, PHP и MySQL. Библия программиста, 2-е издание - М.: Диалектика, 2010. - 170 стр.
- 5 Центр поддержки разработчиков 1С // Электронная версия на сайте <https://dev.1c-bitrix.ru>.

Приложение А

(обязательное)

Техническое задание

А.1.1 Техническое задание на разработку модуля интеграции 1С: Предприятие и интернет-магазина

Настоящее техническое задание распространяется на разработку модуля для интеграции 1С: Предприятие и интернет-магазина на базе OpenCart. Использовать данный модуль будет менеджер, который будет отвечать за наполнение интернет-магазинов соответствующими товарами. Данный модуль позволит значительно сократить время работы с интернет-магазином и частично автоматизировать данный процесс.

А.1.1.1 Назначение

Решение проблемы с выгрузкой большого количества наименований товаров из уже имеющейся базы данных, которую ведет предприятие.

А.1.1.2 Требования к внешнему виду приложения

Модуль должен обладать лаконичным и простым дизайном с минимальным количеством пунктов настройки. Так же должно наблюдаться преемственность модуля с основным дизайном и концепцией интернет-магазина.

А.1.1.3 Требования к функциональным характеристикам

- Выгрузка полной иерархии категорий
- Выгрузка изображений
- Выгрузка описаний товаров
- Выгрузка цен номенклатуры

А.1.1.5 Требования к надежности

Продолжение приложения А

Предусмотреть контроль вводимой информации. Предусмотреть блокировку некорректных действий пользователя при работе с системой. Обеспечить целостность хранимой информации.

Доступ к базе данных сайта из УС обеспечивается аутентификацией со стороны УС, что предотвращает нежелательное подключение к базе данных.

В случае возникновения ошибки при выгрузке данных из УС в интернет-магазин, процесс выгрузки не оборвется, ошибка будет выведена в окно служебных сообщений.

Предусмотреть устойчивость и работоспособность приложения при сбоях и ошибках системы.

А.1.1.6 Требования к составу и параметрам технических средств

- Процессор Intel Pentium IV/Xeon 2,4 ГГц и выше
- Оперативная память 1024 Мб и выше
- Жесткий диск 40Гб и выше

А.1.1.6 Требования к составу и параметрам технических средств

Система должна работать под управлением операционных систем семейства Windows.

Для корректной работы интернет-магазина требуются следующие системные требования:

- Веб-сервер (предпочтительно Apache)
- PHP (не менее 5.2)
- MySQL
- Curl
- mCrypt

Требования к платформе «1С: Предприятие:Управление торговлей»

- Версия платформы не ниже 10.3, 32/64 бит.

– Для работы требуются пользовательские лицензии на платформу 1С: Предприятие в количестве, равном количеству рабочих мест конфигурации.

Требования к платформе «1С: Предприятие:Предприятие»

- Версия платформы 8.3.13, 32/64 бит.

– Для работы требуются пользовательские лицензии на платформу 1С: Предприятие в количестве, равном количеству рабочих мест конфигурации.

- Работа в режиме «Конфигурирование».
- Работа в режиме «Обмен данными с WEB-сайтом».

Продолжение приложения А

А.1.1.7 Требования к программной документации

Разрабатываемые программные модули должны быть задокументированы, т.е. тексты программ должны содержать все необходимые комментарии.

Больше требований к программной документации нет.

Приложение Б (обязательное)

Текст программы

```
//Файл контроллера Controller
<?php
/**
 * @property model_extension_exchange1c
 */
class ControllerExtensionModuleExchange1c extends Controller {
    private $error = array();
    private $module_name = 'Exchange 1C 8.3';

    * Основная функция
    */
    public function index($refresh = false) {
        $data['lang'] = $this->load-
>language('extension/module/exchange1c')
        $this->load->model('tool/image');
        $this->document->setTitle($this->language->get('heading_title'));
        $this->load->model('setting/setting');
        $data['text_info'] = "";
        $this->load->model('extension/exchange1c');
        if (($this->request->server['REQUEST_METHOD'] == 'POST') &&
$this->validate()) {
            // При нажатии кнопки сохранить
            $settings = $this->request->post;
            $settings['exchange1c_version'] = $this->config-
>get('exchange1c_version');
            $settings['exchange_date'] = $this->config-
>get('exchange_date');
            $settings['exchange_statistics'] = $this->config-
>get('exchange_statistics');
            $settings['exchange1c_table_fields'] = $this->config-
>get('exchange1c_table_fields');
            $settings['exchange1c_CMS_version'] = VERSION;
            $settings['exchange1c_table_fields'] = $this-
>model_extension_exchange1c->defineTableFields();
            $this->model_setting_setting->editSetting('exchange1c',
$this->session->data['success'] = $this->language-
>get('text_success');
```

Продолжение приложения Б

```
if (!$refresh) {
    $this->response->redirect($this->url->link('extension/extension', 'token=' . $this->session->data['token'], 'SSL'));
}
$data['text_info'] = "Настройки сохранены";

} else {
    $settings = $this->model_setting_setting->getSetting('exchange1c');
    $data['text_info'] = $this->model_extension_exchange1c->checkUpdates($settings);
    // После обновления прочитаем настройки снова
    //$settings = $this->model_setting_setting->getSetting('exchange1c');
}
if (isset($settings['exchange1c_version'])) {
    $data['version'] = $settings['exchange1c_version'];
} else {
}
$data['url_connect'] = HTTP_CATALOG .
"export/exchange1c.php";
$data['exchange1c_config_icon'] = $this->getParam('config_icon');
if (isset($this->error['warning'])) {
    $data['error_warning'] = $this->error['warning'];
} else {
    $data['error_warning'] = "";
}
// Проверка базы данных
$error = $this->model_extension_exchange1c->checkDB();
if ($error) {
    $data['error_warning'] .= "<br>" . $error;
}
$data['breadcrumbs'] = array();
$data['breadcrumbs'][] = array(
    'text'          => $this->language->get('text_home'),
    'href'          => $this->url->link('common/home',
'token=' . $this->session->data['token'], 'SSL'),
    'separator'    => false
);
$data['breadcrumbs'][] = array(
    'text'          => $this->language->get('text_module'),
```


Продолжение приложения Б

```
'href' => $this->url->link('extension/extension', 'token=' . $this->session->data['token'], 'SSL'),
'separator' => ' :: '
);
$data['breadcrumbs'][] = array(
    'text' => $this->language->get('heading_title'),
    'href' => $this->url->link('extension/module/exchange1c',
'token=' . $this->session->data['token'], 'SSL'),
    'separator' => ' :: '
);
$data['token'] = $this->session->data['token'];
$data['refresh'] = $this->url->link('extension/module/exchange1c/refresh', 'token=' . $this->session->data['token'],
'SSL');
    $data['action'] = $this->url->link('extension/module/exchange1c',
'token=' . $this->session->data['token'], 'SSL');
    $data['cancel'] = $this->url->link('extension/extension', 'token=' .
$this->session->data['token'], 'SSL');
/**
 * ГЕНЕРАЦИЯ ШАБЛОНА
 */
// Магазины
if (isset($this->request->post['exchange1c_stores'])) {
    $data['exchange1c_stores'] = $this->request->post['exchange1c_stores'];
}
else {
    $data['exchange1c_stores'] = $this->config->get('exchange1c_stores');
    if(empty($data['exchange1c_stores'])) {
        $data['exchange1c_stores'][] = array(
            'store_id' => 0,
            'name' => ''
        );
    }
}
// Таблица настроек базы данных
// свойства из торговой системы
$data['exchange1c_properties'] = $this->getParam('properties', array());
// таблица настройки загрузки заказов
$data['exchange1c_order_import'] = $this->getParam('order_import', array());
// таблица настройки выгрузки заказов
```

Продолжение приложения Б

```
$data['exchange1c_order_export'] = $this->getParam('order_export',
array());
    // таблица настройки видов доставки
    $data['exchange1c_order_delivery'] = $this-
>getParam('order_delivery', array());
    // Виды доставки
    $data['order_types_of_delivery'] = array(
        0 => 'в разработке0',
        1 => 'в разработке1',
        2 => 'в разработке2'
    );
    // Генерация опций
    $params = array(
        'username'
        => array('type' => 'input',)
        , 'password'
        => array('type' => 'password',)
        , 'module_status'
=> array('type' => 'radio', 'default' => 1, 'text' => 'on_off')
    );
    foreach ($params as $name => $param) {
        $html = "";
        switch ($param['type']) {
            case 'button':
                $html = $this->htmlButton($name, $param);
                break;
            case 'radio':
                $html = $this->htmlRadio($name, $param);
                break;
            case 'select':
                $html = $this->htmlSelect($name, $param);
                break;
            case 'input':
                $html = $this->htmlInput($name, $param);
                break;
            case 'password':
                $html = $this->htmlInput($name, $param,
'password');
                break;
            case 'textarea':
                $html = $this->htmlTextarea($name,
$param);
```

Продолжение приложения Б

```
break;
                                case 'checkbox':
                                    $html = $this->htmlCheckbox($name,
$param);
                                    break;
                                }
                                if ($html)
                                    $data['html_'. $name] = $this->htmlParam($name,
$html, $param);}
                                // максимальный размер загружаемых файлов
                                $data['lang']['text_max_filesize'] = sprintf($this->language-
>get('text_max_filesize'), @ini_get('max_file_uploads'));
                                $data['upload_max_filesize'] = ini_get('upload_max_filesize');
                                $data['post_max_size'] = ini_get('post_max_size');
                                $links_info = $this->model_extension_exchange1c->linksInfo();
                                $data['links_product_info'] = $links_info['product_to_1c'];
                                $data['links_category_info'] = $links_info['category_to_1c'];
                                $data['links_manufacturer_info'] =
                                $links_info['manufacturer_to_1c'];
                                $data['links_attribute_info'] = $links_info['attribute_to_1c'];
                                // ВЫВОД шаблона
                                $data['header'] = $this->load->controller('common/header');
                                $data['column_left'] = $this->load-
>controller('common/column_left');
                                $data['footer'] = $this->load->controller('common/footer');
                                $this->response->setOutput($this->load-
>view('extension/module/exchange1c', $data));
                                } // index()
                                /**
                                * Установка модуля
                                */
                                public function install() {
                                    $message = "";
                                    $this->load->model('setting/setting');
                                    $settings = $this->model_setting_setting->getSetting('exchange1c',
0);
                                    if (version_compare(VERSION, '2.3') >= 0) {
                                        $this->load->model('extension/extension');
                                        $this->model_extension_extension->install('module',
'exchange1c');
                                    }
                                    $this->load->model('extension/exchange1c');
```

Продолжение приложения Б

```
$this->model_extension_exchange1c->setEvents();
    $module_version = "1.6.4.1";
    // Создадим директорию в кэше
    $cache = DIR_CACHE . 'exchange1c';
    if (!is_dir($cache)) {
        @mkdir($cache);
        @chmod($cache, 0776);
    }
    $this->load->model('setting/setting');
    $settings['exchange1c_version'] =
$module_version;
    $settings['exchange1c_name']
= 'Exchange 1C 8.x for OpenCart
2.x';
    $settings['exchange1c_CMS_version']
= VERSION;
    $settings['exchange1c_seo_category_name'] =
'[category_name]';
    $settings['exchange1c_seo_parent_category_name'] =
'[parent_category_name]';
    $settings['exchange1c_seo_product_name'] =
'[product_name]';
    $settings['exchange1c_seo_product_price'] =
'[product_price]';
    $settings['exchange1c_seo_manufacturer'] =
'[manufacturer]';
    $settings['exchange1c_seo_sku']
= '[sku]';
    $settings['exchange1c_table_fields'] =
$this->model_extension_exchange1c->defineTableFields();
    $this->model_setting_setting->editSetting('exchange1c', $settings);
    // Определение полей таблиц которые могут быть в разных
версиях CMS

    // $this->load->model('extension/module');
    // $this->model_extension_module->addModule('exchange1c',
    //     array(
    //         'version' => $this->module_version,
    //         'name' => $this->module_name
    //     )
    // );
    // Добавляем связь со значением атрибута
```

Продолжение приложения Б

```
$result = $this->db->query("SHOW COLUMNS FROM `". DB_PREFIX .
"product_attribute` WHERE `field` = 'attribute_value_id'");
    if (!$result->num_rows) {
        $this->db->query("ALTER TABLE `". DB_PREFIX .
"product_attribute` ADD `attribute_value_id` INT( 11 ) NOT NULL DEFAULT '0'
AFTER `attribute_id`");
    }
    $result = $this->db->query("SHOW INDEX FROM `".
DB_PREFIX . "product_attribute` WHERE `key_name` = 'value_id'");
    if (!$result->num_rows) {
        $this->db->query("ALTER TABLE `". DB_PREFIX .
"product_attribute` ADD INDEX `value_id` ( `attribute_value_id` )");
    }
    // Добавим индекс, чтобы быстрее работал поиск по
наименованию
    $result = $this->db->query("SHOW INDEX FROM `".
DB_PREFIX . "option_description` WHERE `column_name` = 'name'");
    if (!$result->num_rows) {
        $this->db->query("ALTER TABLE `". DB_PREFIX .
"option_description` ADD INDEX `name_key` ( `name` )");
    }
    // Добавим индекс, чтобы быстрее работал поиск по
наименованию
    $result = $this->db->query("SHOW INDEX FROM `".
DB_PREFIX . "option_value_description` WHERE `column_name` = 'name'");
    if (!$result->num_rows) {
        $this->db->query("ALTER TABLE `". DB_PREFIX .
"option_value_description` ADD INDEX `name_key` ( `name` )");
    }
    // Добавим индекс, чтобы быстрее работал поиск по
наименованию
    $result = $this->db->query("SHOW INDEX FROM `".
DB_PREFIX . "attribute_description` WHERE `column_name` = 'name'");
    if (!$result->num_rows) {
        $this->db->query("ALTER TABLE `". DB_PREFIX .
"attribute_description` ADD INDEX `name_key` ( `name` )");
    }
    // Связь товаров с 1С: Предприятие
    $this->db->query("DROP TABLE IF EXISTS `". DB_PREFIX .
"product_to_1c`");
    $this->db->query(
        "CREATE TABLE `". DB_PREFIX . "product_to_1c` (
```

Продолжение приложения Б

```

        `product_id`                INT(11)                NOT
NULL,
        `guid`
        VARCHAR(64)                NOT NULL,
        `version`
        VARCHAR(32)                NOT NULL,
        UNIQUE KEY `product_link` (`product_id`, `guid`),
        FOREIGN KEY (`product_id`)
        REFERENCES `". DB_PREFIX
."product`(`product_id`)
        ) ENGINE=MyISAM DEFAULT CHARSET=utf8"
    );
    // Связь категорий с 1С: Предприятие
    $this->db->query("DROP TABLE IF EXISTS `". DB_PREFIX .
"category_to_1c`");
    $this->db->query(
        "CREATE TABLE `". DB_PREFIX . "category_to_1c` (
            `category_id`                INT(11)
            NOT NULL,
            `guid`
            VARCHAR(64)                NOT NULL,
            `version`
            VARCHAR(32)                NOT NULL,
            UNIQUE KEY `category_link`
            (`category_id`, `guid`),
            FOREIGN KEY (`category_id`)
            REFERENCES `". DB_PREFIX ."category`(`category_id`)
        ) ENGINE=MyISAM DEFAULT CHARSET=utf8");
    // Свойства из 1С: Предприятие
    $this->db->query("DROP TABLE IF EXISTS `". DB_PREFIX .
"attribute_to_1c`");
    $this->db->query(
        "CREATE TABLE `". DB_PREFIX . "attribute_to_1c` (
            `attribute_id`                INT(11)
            NOT NULL,
            `guid`
            VARCHAR(64)                NOT NULL,
            `type`
            VARCHAR(1)                NOT NULL,
            `version`
            VARCHAR(32)                NOT NULL,

```

Продолжение приложения Б

```
UNIQUE KEY `attribute_link` (`attribute_id`, `guid`),
        FOREIGN KEY (`attribute_id`)
REFERENCES `". DB_PREFIX ."attribute`(`attribute_id`)
        ) ENGINE=MyISAM DEFAULT CHARSET=utf8");
// Значения свойства из 1С: Предприятие
$this->db->query("DROP TABLE IF EXISTS `". DB_PREFIX .
"attribute_value`");
$this->db->query(
        "CREATE TABLE `". DB_PREFIX . "attribute_value` (
                `attribute_value_id`          INT(11)
        NOT NULL AUTO_INCREMENT,
                `attribute_id`                  INT(11)
        NOT NULL,
                `guid`
        VARCHAR(64)          NOT NULL,
                `name`
        VARCHAR(255)         NOT NULL,
        PRIMARY KEY (`attribute_value_id`),
        UNIQUE KEY `attribute_value_key` (`attribute_id`,
`guid`),
        FOREIGN KEY (`attribute_id`)
REFERENCES `". DB_PREFIX ."attribute`(`attribute_id`)
        ) ENGINE=MyISAM DEFAULT CHARSET=utf8"
);
// Привязка опций к товару
$this->db->query("DROP TABLE IF EXISTS `". DB_PREFIX .
"option_to_product`");
$this->db->query(
        "CREATE TABLE `". DB_PREFIX . "option_to_product` (
                `option_id`                      INT(11)
        NOT NULL,
                `product_id`
        VARCHAR(64)          NOT NULL,
        UNIQUE KEY `option_link` (`option_id`,
`product_id`),
        FOREIGN KEY (`option_id`)
REFERENCES `". DB_PREFIX
."option`(`option_id`),
        FOREIGN KEY (`product_id`)
REFERENCES `". DB_PREFIX
."product`(`product_id`)
```

Продолжение приложения Б

```
) ENGINE=MyISAM DEFAULT CHARSET=utf8"
);
// Привязка производителя к каталогу 1С: Предприятие
// В Ид производителя из 1С: Предприятие записывается либо
Ид свойства сопоставленное либо Ид элемента справочника с производителями
$this->db->query("DROP TABLE IF EXISTS `". DB_PREFIX .
"manufacturer_to_1c`");
$this->db->query(
"CREATE TABLE `". DB_PREFIX . "manufacturer_to_1c`
(
        `manufacturer_id`          INT(11)
        NOT NULL,
        `guid`
        VARCHAR(64)
        NOT NULL,
        UNIQUE KEY `manufacturer_link`
(`manufacturer_id`, `guid`),
        FOREIGN KEY (`manufacturer_id`)
        REFERENCES `". DB_PREFIX . "manufacturer`(`manufacturer_id`)
) ENGINE=MyISAM DEFAULT CHARSET=utf8"
);
// Привязка магазина к каталогу в 1С: Предприятие
$this->db->query("DROP TABLE IF EXISTS `". DB_PREFIX .
"store_to_1c`");
$this->db->query(
"CREATE TABLE `". DB_PREFIX . "store_to_1c` (
        `store_id`
        INT(11)
        NOT NULL,
        `guid`
        VARCHAR(64)
        NOT NULL,
        UNIQUE KEY `store_link` (`store_id`, `guid`),
        FOREIGN KEY (`store_id`)
        REFERENCES `". DB_PREFIX . "store`(`store_id`)
) ENGINE=MyISAM DEFAULT CHARSET=utf8"
);
// Остатки товара
$this->db->query("DROP TABLE IF EXISTS `".
DB_PREFIX . "product_quantity`");
$this->db->query(
"CREATE TABLE IF NOT EXISTS `". DB_PREFIX .
"product_quantity` (
        `product_quantity_id`     INT(11)
        NOT NULL AUTO_INCREMENT,
```


Продолжение приложения Б

```

        `product_id`                INT(11)                NOT
NULL,
                                `product_feature_id`          INT(11)
        DEFAULT 0 NOT NULL,
                                `warehouse_id`                INT(11)
        DEFAULT 0 NOT NULL,
                                `quantity`
DECIMAL(10,3)                DEFAULT 0,
                                PRIMARY KEY (`product_quantity_id`),
                                UNIQUE KEY `product_quantity_key`
(`product_id`, `product_feature_id`, `warehouse_id`),
                                FOREIGN KEY (`product_id`)
REFERENCES ``. DB_PREFIX . "product" (`product_id`),
                                FOREIGN KEY (`product_feature_id`)
REFERENCES ``. DB_PREFIX . "product_feature" (`product_feature_id`),
                                FOREIGN KEY (`warehouse_id`)
REFERENCES ``. DB_PREFIX . "warehouse" (`warehouse_id`),
                                INDEX (`product_id`)
                                ) ENGINE=MyISAM DEFAULT CHARSET=utf8"
);
// Характеристики товара
// Если характеристики не используются, эта таблица будет
пустая
$this->db->query("DROP TABLE IF EXISTS ``. DB_PREFIX .
"product_feature`");
$this->db->query(
        "CREATE TABLE ``. DB_PREFIX . "product_feature` (
                `product_feature_id`                INT(11)
        NOT NULL AUTO_INCREMENT,
                `product_id`                        INT(11)
        NOT NULL,
                `name`
        VARCHAR(256)                NOT NULL,
                `ean`
        VARCHAR(14)                NOT NULL,
                `sku`
        VARCHAR(128)                NOT NULL,
                `guid`
        VARCHAR(64)                NOT NULL,
                PRIMARY KEY (`product_feature_id`),
                UNIQUE KEY `product_feature_key` (`product_id`,
`guid`),

```

Продолжение приложения Б

```
INDEX (`product_id`)
    ) ENGINE=MyISAM DEFAULT CHARSET=utf8"
);
// Товарные категории
// Если товарные категории не используются, эта таблица будет
пустая
$this->db->query("DROP TABLE IF EXISTS `". DB_PREFIX .
"product_category`");
$this->db->query(
    "CREATE TABLE `". DB_PREFIX . "product_category` (
        `product_category_id`          INT(11)
        NOT NULL AUTO_INCREMENT,
        `parent_id`                    INT(11)
        NOT NULL,
        `name`
        VARCHAR(256)                    NOT NULL,
        `guid`
        VARCHAR(64)                    NOT NULL,
        PRIMARY KEY (`product_category_id`),
        UNIQUE KEY `product_category_key`
        (`product_category_id`, `guid`),
        INDEX (`parent_id`)
    ) ENGINE=MyISAM DEFAULT CHARSET=utf8"
);
// Привязка опций к свойствам для предложений в 1С:
Предприятие
$this->db->query("DROP TABLE IF EXISTS `". DB_PREFIX .
"option_to_1c`");
$this->db->query(
    "CREATE TABLE `". DB_PREFIX . "option_to_1c` (
        `option_id`                    INT(11)
        NOT NULL,
        `guid`
        VARCHAR(64)                    NOT NULL,
        UNIQUE KEY `option_key` (`option_id`, `guid`),
        FOREIGN KEY (`option_id`)
        REFERENCES `". DB_PREFIX
."option`(`option_id`)
    ) ENGINE=MyISAM DEFAULT CHARSET=utf8"
);
// Значения характеристики товара(доп. значения)
```

Продолжение приложения Б

```
$this->db->query("DROP TABLE IF EXISTS `". DB_PREFIX .
"product_feature_value`");
    $this->db->query(
        "CREATE TABLE `". DB_PREFIX .
"product_feature_value` (
            `product_feature_id`          INT(11)
            NOT NULL,
            `product_option_id`          INT(11)
            NOT NULL,
            `product_id`                  INT(11)
            NOT NULL,
            `product_option_value_id`    INT(11)
            NOT NULL,
            UNIQUE KEY `product_feature_value_key`
            (`product_feature_id`, `product_id`, `product_option_value_id`),
            FOREIGN KEY (`product_feature_id`)
            REFERENCES `". DB_PREFIX . "product_feature`(`product_feature_id`),
            FOREIGN KEY (`product_option_id`)
            REFERENCES `". DB_PREFIX . "product_option`(`product_option_id`),
            FOREIGN KEY (`product_id`)
            REFERENCES `". DB_PREFIX .
"product`(`product_id`),
            FOREIGN KEY (`product_option_value_id`)
            REFERENCES `". DB_PREFIX .
"product_option_value`(`product_option_value_id`)
        ) ENGINE=MyISAM DEFAULT CHARSET=utf8"
    );

    // Цены, если характеристики не используются, эта таблица
    // будет пустая
    $this->db->query("DROP TABLE IF EXISTS `". DB_PREFIX .
"product_price`");
    $this->db->query(
        "CREATE TABLE `". DB_PREFIX . "product_price` (
            `product_price_id`          INT(11)
            NOT NULL AUTO_INCREMENT,
            `product_id`                  INT(11)
            NOT NULL,
            `product_feature_id`          INT(11)
            NOT NULL DEFAULT '0',
            `customer_group_id`          INT(11)
            NOT NULL DEFAULT '0',
```

Продолжение приложения Б

```
        `price`                                DECIMAL(15,4)      NOT NULL
DEFAULT '0',

        PRIMARY KEY (`product_price_id`),
        UNIQUE KEY `product_price_key` (`product_id`,
`product_feature_id`, `customer_group_id`),
        FOREIGN KEY (`product_id`)
        REFERENCES ``. DB_PREFIX .
"product"(`product_id`),
        FOREIGN KEY (`product_feature_id`)
        REFERENCES ``. DB_PREFIX . "product_feature"(`product_feature_id`),
        INDEX (`product_id`)
    ) ENGINE=MyISAM DEFAULT CHARSET=utf8"
);
} // install()
/**
 * Алгоритм описан https://dev.1c-bitrix.ru/api\_help/sale/algorithms/data\_2\_site.php
 * Авторизация на сайте
 */
public function modeCheckauth() {
    if (!$this->checkAccess(true))
        exit;
    $auth_user = "";
    $auth_pw = "";
    // Определение авторизации на сервере
    if (!isset($_SERVER['PHP_AUTH_USER'])) {
        if (isset($_SERVER["REMOTE_USER"]) &&
isset($_SERVER["REDIRECT_REMOTE_USER"])){
            $remote_user = $_SERVER["REMOTE_USER"] ?
$_SERVER["REMOTE_USER"] : $_SERVER["REDIRECT_REMOTE_USER"];
        } elseif
(isset($_SERVER["REDIRECT_HTTP_AUTHORIZATION"])) {
            $remote_user =
$_SERVER["REDIRECT_HTTP_AUTHORIZATION"];
        }
        if (isset($remote_user)) {
            $strTmp = base64_decode(substr($remote_user,6));
            if($strTmp)
                list($auth_user, $auth_pw) = explode(':',
$strTmp);
        } else {
```

Продолжение приложения Б

```
$this->echo_message(0, 'Auth Method Failed, check var $_SERVER');
    exit;
    }
    } else {
        $auth_user = $_SERVER['PHP_AUTH_USER'];
        $auth_pw = $_SERVER['PHP_AUTH_PW'];
    }
    // Авторизуем
    if (($this->config->get('exchange1c_username') != '') &&
($auth_user != $this->config->get('exchange1c_username'))) {
        $this->echo_message(0, "Incorrect login");
    }
    if (($this->config->get('exchange1c_password') != '') && ($auth_pw
!= $this->config->get('exchange1c_password'))) {
        $this->echo_message(0, "Incorrect password");
        exit;
    }
    echo "success\n";
    echo "key\n";
    echo md5($this->config->get('exchange1c_password')) . "\n";
} // modeCheckauth()
/**
 * Проверка существования каталогов
 */
private function checkDirectories($name) {
    $path = DIR_IMAGE;
    $dir = explode("/", $name);
    for ($i = 0; $i < count($dir)-1; $i++) {
        $path .= $dir[$i] . "/";
        if (!is_dir($path)) {
            $error = "";
            @mkdir($path, 0775) or die ($error = "Ошибка
создания директории " . $path . "");
            if ($error)
                return $error;
        }
    }
    return "";
} // checkDirectories()
/**
 * Распаковываем картинки
 */
```

Продолжение приложения Б

```
private function extractImage($zipArc, $zip_entry, $name) {
    $error = "";
    if (substr($name, -1) == "/") {
        // проверим каталог
        if (is_dir(DIR_IMAGE . $name)) {
        } else {
            @mkdir(DIR_IMAGE . $name, 0775) or die ($error
= "Ошибка создания директории " . DIR_IMAGE . $name . "");
            if ($error) return $error;
        }
    } elseif (zip_entry_open($zipArc, $zip_entry, "r")) {
        $dump = zip_entry_read($zip_entry,
zip_entry_filesize($zip_entry));
        $error = $this->checkDirectories($name);
        if ($error)
            return $error;
        if (is_file(DIR_IMAGE . $name)) {
            // Если файл существует
            $size_dump = strlen($dump);
            $size_file = filesize(DIR_IMAGE . $name);
            if ($size_dump != $size_file) {
                // файл был изменен, нужно заменить
                $fd = @fopen(DIR_IMAGE . $name, "wb");
                if ($fd === false) {
                    return "Ошибка записи файла: " .
DIR_IMAGE . $name . "";
                }
                fwrite($fd, $dump);
                fclose($fd);
                if ($this->config-
>get('exchange1c_log_level') == 3) {
                    // проверим записанный размер
                    $new_size_file =
                    filesize(DIR_IMAGE . $name);
                    if ($size_file != $new_size_file) {
                    }
                }
            } else {
                // для безопасности проверим, не является ли
                этот файл php
                $pos = strpos($dump, "<?php");
                if ($pos !== false) {
```

Продолжение приложения Б

```
$this->echo_message("[!] ВНИМАНИЕ Файл '" . $name . "' является PHP
скриптом и не будет записан!");
    } else {
        $fd = @fopen(DIR_IMAGE . $name, "wb");
        if ($fd === false) {
            return "Ошибка создания файла: "
. DIR_IMAGE . $name . ", проверьте права доступа!";
        }
        fwrite($fd, $dump);
fclose($fd);
// для безопасности проверим, является
ли этот файл картинкой
// $image_info =
getimagesize(DIR_IMAGE.$name);
// if ($image_info == NULL) {
//     $this->log("[!] ВНИМАНИЕ Файл
'" . $name . "' не является картинкой, и будет удален!");
//     unlink(DIR_IMAGE.$name);
// }
    }
}
zip_entry_close($zip_entry);
}
return $error;
} // extractImage()
/**
 * Распаковываем XML
 */
private function extractXML($zipArc, $zip_entry, $name, &$xmlFiles) {
    $error = "";
    $cache = DIR_CACHE . 'exchange1c/';
    if (substr($name, -1) == "/") {
        // это директория
        if (is_dir($cache . $name)) {
        } else {
            @mkdir($cache . $name, 0775) or die ($error =
"Ошибка создания директории '" . $cache . $name . "'");
            if ($error) return $error;
        }
    } elseif (zip_entry_open($zipArc, $zip_entry, "r")) {
        $dump = zip_entry_read($zip_entry,
zip_entry_filesize($zip_entry));
```

Продолжение приложения Б

```
// Удалим существующий файл
if (file_exists($cache . $name)) {
    unlink($cache . $name);
}
// для безопасности проверим, является ли этот файл
XML
$str_xml = substr($dump, 1, 35);
$path_obj = explode('/', $name);
$filename = array_pop($path_obj);
$path = "";
// Если есть каталоги, нужно их создать
foreach ($path_obj as $dir_name) {
    if (!is_dir($cache . $path . $dir_name)) {
        @mkdir($cache . $path . $dir_name, 0775)
or die ($error = "Ошибка создания директории " . $cache . $path . $dir_name . "");

    } else {
        $this->echo("Папка уже существует: ");
    }
    $path .= $dir_name . "/";
}
if (stripos($str_xml, "?xml") === false) {
    $this->echo("[!] ВНИМАНИЕ Файл не является
XML файлом и не будет записан!");
} elseif ($fd = @fopen($cache . $path . $filename, "wb")) {
    $xmlFiles[] = $path . $filename;
    fwrite($fd, $dump);
    fclose($fd);
} else {
}
zip_entry_close($zip_entry);
}
$this->echo("Завершена распаковка XML");
return "";
} // extractXML()

/**
 * Распаковываем ZIP архив
 * Папка с картинками распаковывается в папку что указана в архиве
 */
private function extractZip($zipFile, &$error) {
```


Продолжение приложения Б

```
$xmlFiles = array();
$imgFiles = 0;
$zipArc = zip_open($zipFile);
if (is_resource($zipArc)) {
    while ($zip_entry =
zip_read($zipArc)) {
        $name = zip_entry_name($zip_entry);
        $pos = stripos($name, 'import_files');
        if ($pos !== false) {
            $error = $this->extractImage($zipArc,
$zip_entry, substr($name, $pos));
            if ($error) return $xmlFiles;
            $imgFiles ++;
        } else {
            $error = $this->extractXML($zipArc,
$zip_entry, $name, $xmlFiles);
            if ($error) return $xmlFiles;
        }
    }
}
else {
    return $xmlFiles;
}
zip_close($zipArc);
return $xmlFiles;
} // extractZip()

/**
 * Определяет тип файла по наименованию
 */
public function detectFileType($fileName) {
    $types = array('import', 'offers', 'prices', 'rests');
    foreach ($types as $type) {
        $pos = stripos($fileName, $type);
        if ($pos !== false)
            return $type;
    }
    return "";
} // detectFileType()

public function modeCatalogInit() {
    if (!$this->checkAuthKey()) {
```

Продолжение приложения Б

```
echo "failure";
        exit;
    }
    $result = $this->modeInit();
    echo $result[0] . "\n";
    echo $result[1] . "\n";
    } // modeCatalogInit()

/**
 * Обрабатывает команду инициализации
 */
public function modeSaleInit() {
    if (!$this->checkAuthKey()) {
        echo "failure";
        exit;
    }
    $result = $this->modeInit();
    echo $result[0] . "\n";
    echo $result[1] . "\n";
} // modeSaleInit()

/**
 * Обрабатывает команду инициализации
 */
public function modeInit() {
    $result = array();
    $zip_support = class_exists('ZipArchive') ? true : false;
    if ($zip_support) {
        $result[0] = $this->config->get('exchange1c_file_zip') == 1 ?
"zip=yes" : "zip=no";
    } else {
        $result[0] = "zip=no";
    }
    $manual_size = $this->formatSize($this->config-
>get('exchange1c_file_max_size'));
    $post_max_size = $this->getPostMaxFileSize();
    if ($this->config->get('exchange1c_file_max_size') &&
$manual_size <= $post_max_size) {
        $result[1] = "file_limit=" . $manual_size;
    } else {
        $result[1] = "file_limit=" . $post_max_size;
```

Продолжение приложения Б

```
}
    return $result;
} // modeInit()
/**
 * Проверяет наличие куки ключа
 */
private function checkAuthKey($echo=true) {
    if (!isset($this->request->cookie['key'])) {
        if ($echo) $this->echo_message(0, "no cookie key\n");
        return false;
    }
    if ($this->request->cookie['key'] != md5($this->config-
>get('exchange1c_password'))) {
        if ($echo) $this->echo_message(0, "Session error\n");
        return false;
    }
    return true;
}

/**
 * Переводит значение из килобайт, мегабайт и гигабайт в байты
 */
private function formatSize($size) {
    if (empty($size)) {
        return 0;
    }
    $type = $size{strlen($size)-1};
    if (!is_numeric($type)) {
        $size = (integer)$size;
        switch ($type) {
            case 'K': $size = $size*1024;
                break;
            case 'M': $size = $size*1024*1024;
                break;
            case 'G': $size = $size*1024*1024*1024;
                break;
        }
        return $size;
    }
    return (int)$size;
} // formatSize()
/**
```

Продолжение приложения Б

```
* Обработывает загруженный файл на сервер
*/
private function modeFile($mode, &$error) {
    $xmlfiles = array();
    if (!$this->checkAuthKey()) exit;
    $cache = DIR_CACHE . 'exchange1c/';

    // Проверяем на наличие каталога
    if(!is_dir($cache)) mkdir($cache);

    // Проверяем на наличие имени файла
    if (isset($this->request->get['filename'])) {
        $uplod_file = $cache . $this->request->get['filename'];
    }
    else {
        $error = "modeFile(): No file name variable";
        return false;
    }
    // Проверяем XML или изображения
    if (strpos($this->request->get['filename'], 'import_files') !== false) {
        $cache = DIR_IMAGE;
        $uplod_file = $cache . $this->request->get['filename'];
        $this->checkUploadFileTree(dirname($this->request-
>get['filename']), $cache);
    }
    // Получаем данные
    $data = file_get_contents("php://input");
    if ($data !== false) {
        // Записываем в файл
        $filesize = file_put_contents($uplod_file, $data,
FILE_APPEND | LOCK_EX);
        if ($filesize) {
            chmod($uplod_file, 0664);
            $xmlfiles = $this->extractZip($uplod_file, $error);
            if ($error) {
                $this->echo_message(0, "modeFile(): Error
extract file: " . $uplod_file);
            }
            if ($this->config-
>get('exchange1c_not_delete_files_after_import') != 1) {
                unlink($uplod_file);
            }
        }
        return false;
    }
}
```

Продолжение приложения Б

```
};
        } else {
            $this->echo_message(0, "modeFile(): Error create
file");
        }
    }
    else {
        $this->echo_message(0, "modeFile(): Data empty");
    }
    return $xmlfiles;
} // modeFile()
/**
 * Обрабатывает загруженный файл на сервер
 */
public function modeFileCatalog() {
    $error = "";
    $this->modeFile('catalog', $error);
    if ($error) {
        $this->echo_message(0, $error);
    } else {
        $this->echo_message(1, "Successfully import catalog ");
    }
} // modeFileCatalog()

/**
 * Обрабатывает *.XML файл
 */
public function modeImport($manual = false) {
    $cache = DIR_CACHE . 'exchange1c/';
    if(!is_dir($cache)) mkdir($cache);
    // Определим имя файла
    if ($manual)
        $importFile = $manual;
    elseif (isset($this->request->get['filename']))
        $importFile = $cache . $this->request->get['filename'];
    else {
        if (!$manual) $this->echo_message(0, "No import file
name");
    }

    $this->load->model('extension/exchange1c');
    // Загружаем файл
```

Продолжение приложения Б

```
$error = $this->model_extension_exchange1c->importFile($importFile, $this->detectFileType($importFile));

    if ($this->config->get('exchange1c_not_delete_files_after_import')
    != 1) {
        @unlink($importFile);
    }
    if ($error) {
        if (!$manual) {
            $this->echo_message(0, 'ERROR: ' . $error);
        }
        return $error;
    } else {
        if (!$manual) {
            $this->echo_message(1, "Successfully processed
file: " . $importFile);
        }
    }
    return "";
} // modeImport()
/**
 * Формирует архив модуля для инсталляции
 */
public function modeExportModule() {
    if ($this->config->get('exchange1c_export_module_to_all') != 1) {
        if (!$this->checkAccess(true)) {
            echo "<br />\n";
            echo "Экспорт модуля " . $this->module_name . "
для IP " . $_SERVER['REMOTE_ADDR'] . " в данный момент запрещен!";
            return false;
        }
    }
}
```