**ТЕХНИЧЕСКОЕ ЗАДАНИЕ ПРОЕКТА**

«Система агрегации сведений о книгах в книжных магазинах»

**Глоссарий**

|  |  |
| --- | --- |
| **Термин** | **Определение** |
| Валидация данных | Проверка на корректность, полноту и непротиворечивость входных, выходных и обрабатываемых данных. |
| Децентрализация | Тип архитектуры программных систем без создания выделенного единственного центра. |
| WEB-интерфейс | Интерфейс пользователя, предоставляемой системой через Web-браузер. |
| Сервис | Идентифицируемая веб-адресом программная система со стандартизированными интерфейсами. |
| JSON | Текстовый формат обмена данными, основанный на JavaScript. Как и многие другие текстовые форматы, JSON легко читается людьми. |

**Принятые сокращения**

|  |  |
| --- | --- |
| **Сокращение** | **Расшифровка** |
| ОС | Операционная система |
| СПО | Специальное программное обеспечение |
| БД | База данных |
| СУБД | Система управления базами данных |

**Введение**

Данное техническое задание составляется для проектирования СПО «Система агрегации сведений о книгах в книжных магазинах» (далее – СПО). Техническое задание выполняется в соответствии со стандартом ГОСТ 34.602-89 «Техническое задание на создание автоматизированной системы».

**Краткое описание предметной области**

Для поиска необходимой книги по определенному материалу занимает большое количество времени, т.к. в одном конкретном книжном магазине не может быть всех тех книг, которые могут понадобиться тому или иному читателю. Благодаря наличию веб-порталов у книжных магазинов несколько упрощается поиск книг.

Для более эффективного поиска необходимо объединить сведения о наличии того или иного издания в книжных магазинах в единую систему, через которую пользователь сможет быстро определять, где находится интересующая его книга.

**Существующие аналоги**

В качестве аналогов могут выступать такие крупные системы, как avito.ru, в которой агрегируется информация не только о книжных магазинах, но и о организациях, оказывающих другого рода услуги, объявления частных лиц.

Данный проект должен иметь следующие преимущества перед существующими аналогами:

* предоставлять доступ автоматической регистрации изданий для книжных магазинов;
* содержать актуальную информацию о наличии изданий.

**Описание системы**

Проект должен представлять собой портал, на котором регистрируются пользователи и книжны магазины. Магазин сохраняет на портале о своем местоположении и имеющихся в наличии книг с указанием стоимости. На основании данной информации клиенты, посещающие портал, производят поиск интересующих их книг.

Схема предметной области изображена на Рис. П1.1.

|  |
| --- |
|  |
| Рис. П1.1 Схема предметной области |

**Основания для разработки**

Разработка ведется в рамках выполнения лабораторных работ по курсу Методология программной инженерии на кафедре «Программное обеспечение ЭВМ и информационные технологии» факультета «Информатика и системы управления» МГТУ им. Н. Э. Баумана.

**Назначения разработки**

Главное назначение разрабатываемого портала — облегчение поиска книжного магазина пользователем, в котором имеется в наличии интересующее клиента издание. Портал должен учитывать все пожелания со стороны клиента и со стороны книжных магазинов. Клиент выбирает тот книжный магазин, в котором есть интересующего издание по приемлемой для него цене. Книжные магазины публикуют на данном портале информацию об имеющихся у них в наличие книгах с целью увеличения прибыли и области сбыта.

**Требования к системе**

1. Система должна обеспечивать хранение информации о зарегистрированных книжных магазинах, о книгах, которые в них есть.
2. Система должна обеспечивать хранение детальной информации о книгах, зарегистрированных в системе.
3. Система должна быть децентрализованной и не выходить из строя в случае выхода из строя одной из подсистем.
4. Система должна предоставлять пользователю WEB-интерфейс для выбора необходимых книг и оформления заказа.

**Требования к функциональным характеристикам**

1. Время отклика системы на запрос пользователя не должно превышать 10 секунд.
2. Время отклика системы на запрос «Добавить книгу в заказ» не должно превышать 2 секунд.
3. Обеспечить заданные временные характеристики в случае наличия 1000 книг в базе данных.

**Функциональные требования к системе с точки зрения пользователям**

Портал должен обеспечивать реализацию следующих функций:

1. Система должна обеспечивать регистрацию пользователей.
2. Система должна обеспечивать аутентификацию пользователей.
3. Система должна обеспечить разделение пользователей на три роли:

* книжный магазина;
* клиент;
* администратор.

1. Система должна предоставить **клиенту** следующие функции:

* просмотр информации об имеющихся книгах;
* просмотр информации о зарегистрированных книжных магазинах;
* выбор книг и оформление заказа.

1. Система должна предоставить **книжному магазину** следующие функции:

* ввод информации о магазине;
* регистрация книг с указанием цены.

1. Система должна предоставить **администратору** следующие функции:

* неограниченные возможности по изменению контента на портал.

**Входные параметры системы**

1. Система должна содержать следующую информацию о пользователях:

* учетное имя пользователя (необходимо для идентификации);
* пароль от учетной записи;
* имя пользователя;
* фамилия пользователя;
* время добавления пользователя в систему.

1. Система должна содержать следующую информацию о книжном магазине:

* название магазина;
* адрес магазина;
* список книг, которые присутствуют в магазине.

1. Система должна содержать следующую информацию о книге:

* авторы книги;
* название книги;
* год издания;
* издательство;
* краткое описание.

1. Система должна содержать следующую информацию о заказе:

* пользователь, оформивший заказ;
* список книг в заказе;
* состояние выполнения заказа.

**Выходные параметры системы**

Выходными параметрами системы являются веб-страницы. Они должны содержать следующую информацию:

* Список книг с возможностью добавления/удаления книги из заказа.
* Состояние исполнения заказа.

**Требования к составу и параметрам технических средств**

Все серверные приложения должны потреблять суммарно не более 2 Гб оперативной памяти и работать на сервере с процессором Intel i7 2.4 GHz и жестким диском SATA 100 IOPS 100 Гб.

**Требования к надежности**

Система должна работать в соответствии с данным техническим заданием без рестарта.

**Требования к документации**

Исполнитель должен подготовить и передать Заказчику следующие документы:

* руководство по развертыванию Системы;
* руководство администратора Системы;
* руководство для книжного магазина по использованию Системы;
* руководство для клиента по использованию Системы.

**Топология системы**

На Рис. П1.2 приведена один из возможных вариантов топологии разрабатываемой распределенной системы.

|  |
| --- |
|  |
| Рис. П1.2 Топология СПО «Система заказа книг из библиотек» |

Система будет состоять из web-сервера и 4 подсистем:

1. подсистема хранения информации о книгах
2. подсистема хранения информации о зарегистрированных книжных магазинах
3. подсистема хранения информации о заказах
4. подсистема хранения информации о сессиях

**Подсистема для хранения сведений о книгах** отвечает за хранение информации о зарегистрированных книжными магазинами книгах.

**Подсистема для хранения сведений о книжных магазинах** отвечает за хранение информации о зарегистрированных в системе книжных магазинах.

**Подсистема для хранения заказов** отвечает за хранение информации о совершенных клиентами заказов.

**Подсистема для хранения сессии** отвечает за сессию пользователей портала и реализует следующие функции:

* + регистрация пользователя;
  + аутентификация (проверка сессии) пользователя;
  + авторизация пользователя (вход, или «логин»);
  + выход из сессии («логаут»).

**Web-сервер** принимает запросы от пользователей по протоколу HTTP и анализирует их. На основе проведенного анализа он выполняет запросы к подсистемам, агрегирует их ответы и отсылает ответ пользователю.

**Требования по реализации со стороны заказчика**

1. Все бекенды и фронтенд должны быть запущены изолированно друг от друга.
2. Требуется использовать СОА (сервис-ориентированную архитектуру).
3. Разработка серверных приложений может осуществляться с использованием разных фреймворков для разных бекендов, поскольку бекенды должны быть слабо связаны друг с другом по требованию сервисно-ориентированного подхода к разработке.
4. Необходимо реализовать один веб-интерфейс для фронтенда. Интерфейс должен быть доступен через тонкий клиент – браузер.
5. Серверы бекендов недоступны пользователю, это реализуется их расположением во внутренней сети.
6. Валидация входных данных должна производиться и на стороне пользователя с помощью JavaScript скриптов, и на стороне фронтенда. Бекенды не должны валидировать входные данные, так как пользователь не может к ним обращаться напрямую, бекенды должны получать уже отфильтрованные входные данные от фронтенда.

**Функциональные требования по подсистемам**

1. ***Фронтенд*** – это серверное приложение при разработки которого необходимо учитывать следующие факторы:

* фронтенд должен принимать запросы по протоколу HTTP и формировать ответы пользователям портала в формате HTML;
* в зависимости от типа запроса фронтенд должен отправлять последовательные запросы в соответствующие бекенды;
* запросы к бекендам необходимо осуществляет по протоколу HTTP. Данные необходимо передавать в формате JSON. Данный текстовый формат обмена данными удобен для чтения;
* целесообразно использовать библиотеку Bootstrap v3 для создания HTML-верстки.

1. ***Сессионный бекенд*** – это серверное приложение, которое должно отвечать следующим требованиям по разработке.

* Сессионный бекенд должен принимать и возвращать данные в формате JSON по протоколу HTTP.
* Выполнять авторизацию пользователей, проверять и удалять сессию, а также регистрировать пользователей.
* База данных, содержащая информацию о сессиях, должна находиться на этом же сервере. Доступ к СУБД должен осуществляться по протоколу TCP.

1. ***Бекенд заказов*** – это серверное приложение, которое должно отвечать следующим требованиям по разработке.

* Бекенд должен принимать и возвращать данные в формате JSON по протоколу HTTP.
* Сохранять информацию о сформированных заказах и их состоянии завершенности.
* База данных, содержащая информацию о заказах, должна находиться на этом же сервере. Доступ к СУБД должен осуществляться по протоколу TCP.

1. ***Бекенд книжных магазинов*** – это серверное приложение, которое должно отвечать следующим требованиям по разработке.

* Бекенд должен принимать и возвращать данные в формате JSON по протоколу HTTP.
* Сохранять информацию о зарегистрированных книжных магазинах.
* База данных, содержащая информацию о заказах, должна находиться на этом же сервере. Доступ к СУБД должен осуществляться по протоколу TCP.

1. ***Бекенд книг*** – это серверное приложение, которое должно отвечать следующим требованиям по разработке.

* Бекенд должен принимать и возвращать данные в формате JSON по протоколу HTTP.
* Сохранять информацию о зарегистрированных книгах.
* База данных, содержащая информацию о заказах, должна находиться на этом же сервере. Доступ к СУБД должен осуществляться по протоколу TCP.