# 1 Техническое задание

## Глоссарий

|  |  |
| --- | --- |
| **Термин** | **Определение** |
| Валидация данных | Проверка на корректность, полноту и непротиворечивость входных, выходных и обрабатываемых данных |
| WEB-интерфейс | Интерфейс пользователя, предоставляемой системой через Web-браузер |
| Хеширование | Преобразование входного массива данных произвольной длины в выходную битовую строку фиксированной длины |

## Принятые сокращения

|  |  |
| --- | --- |
| **Сокращение** | **Расшифровка** |
| ОС | Операционная система |
| СПО | Специальное программное обеспечение |

## Введение

Данное техническое задание разработано на программный комплекс «Распределенная система обработки информации по заказу электроники из интернет-магазинов», в дальнейшем именуемое система заказа электроники.

Данное техническое задание формируется на основе ГОСТ 9.201-78 «Техническое задание. Требования к содержанию и оформлению».

### Краткое описание предметной области

Проблема просмотра и выгодного заказа электронных товаров в интернете в последнее время является актуальной. Это связано с тем, что каждый интернет-магазин продает схожий набор товаров по разной цене. Также, помимо цены, разнится и цена за доставку. Покупатель, заказывая товар из интернет-магазина, надеется, что товар придет ему соответствующего качества и за минимальную цену, но часто выходит, что это не так, так как цена за этот же товар в другом интернет-магазине ниже.

### Существующие аналоги

### На данный момент существует сервис “Яндекс.Маркет”, на котором размещены товары и магазины, перейдя на сайт которых можно заказать то или иное устройство, но тогда, когда сайт магазина не доступен, заказать товар невозможно.

## 1.1 Описание системы

Для систематизации построения разрабатываемая система должна быть разделена на сервисы. Разбиение на сервисы призвано разделить и изолировать отдельные участки системы, структурировать построение, упростить формирование требований к отдельным частям системы.

При построении должны быть реализованы следующие системы:

1.       Подсистема “Фронтенд”

2.       Подсистема “Сервис сессии”

3.       Подсистема “Сервис логики”

4.       Подсистема “Сервис по работе с покупателем”

5.       Подсистема “Сервис по работе с фирмами”

6.       Подсистема “Сервис обработки заказов”

7.       Подсистема “Сервис по работе с товарами”

Подсистема “Фронтенд” отвечает за получение входной информации от пользователя (покупателя или фирмы) и обработку полученной информации.

Подсистема “Сервис сессии” предназначен для создания и проверки сессии пользователя. Также через этот сервис осуществляется доступ к базам данных «Фирмы» и «Покупатели», в которых хранятся логины и пароли пользователей.

Подсистема “Сервис по работе с покупателем” предназначен для обработки запросов пользователя-покупателя.

Подсистема “Сервис по работе с фирмами” предназначен для обработки запросов пользователя-фирмы. Также через этот сервис осуществляется доступ к базе данных «Фирма – Товар».

Подсистема “Сервис по работе с товарами” предназначен для доступа к базе данных «Товары».

Подсистема “Сервис обработки заказов” предназначен для доступа к базам данных «Заказы» и «Состав заказов».

Подсистема “Сервис логики” предназначен для обработки запросов и данных, полученных от других сервисов.

Рисунок 1 Архитектура разрабатываемой системы

## 1.2 Основания для разработки

Разработка ведется в рамках выполнения лабораторных работ по курсу «Методология программной инженерии» на основании утверждённого учебного плана. Разработка РСОИ должна включать в себя следующие этапы:

1. Разработка и утверждение технического задания, протокола о принципах взаимодействия систем и уточнения основных особенностей функционирования РСОИ.
2. Разработка программного обеспечения, в том числе разработку релизов. В первом релизе должны быть реализованы функции, которые позволяют оценить правильность стратегии в выборе архитектурных решений.
3. Подготовка сопроводительной документации
4. Внедрение системы
5. Тестирование в реальных условиях

## 1.3 Требования к системе

### 1.3.1 Требования к РСОИ

1. Разрабатываемое программное обеспечение должно обеспечивать функционирование системы в режиме 24/7/365.
2. Время восстановления системы после сбоя не должно превышать 3-х часов.
3. Система должна поддерживать возможность «горячего» переконфигурирования системы.
4. Система должна обеспечивать хранение заказов, данных о пользователях и товарах
5. Система должна обеспечивать децентрализацию сбора информации с помощью формирования вынесенных локальных серверов сбора информации, которые, в свою очередь, передают информацию в центральный сервер
6. Система должна обеспечивать визуализацию данных о товарах и заказах
7. Система должна поддерживать добавление новых пользователей
8. Система не должна выходить из строя при выходе из строя одной из подсистем
9. Обмен информацией в системе должен производиться исходя из предположения, что каналы связи небезопасны и ненадежны

### 1.3.2 Требования к функциональным характеристикам

1. Время реактивности системы -3-5сек. (по требованию заказчика);
2. Время выдачи списка предложений от фирмы - не более 7 сек.;
3. Синхронные запросы между подсистемами не должны превышать 10 секунд;
4. Время отклика системы на запрос пользователя не должно превышать 20 секунд;
5. Время отклика системы на запрос «Зарегистрировать пользователя в системе» не должно превышать 21 секунды;
6. Обеспечить заданные временные характеристики для 20 одновременно подключённых пользователей.

### 1.3.3 Требования к реализации

1. Система должна обеспечивать возможность запуска всех компонентов как сервисов ОС;
2. Интерфейсы управления должны быть реализованы в виде WEB-интерфейсов;
3. Пароли учетных записей должны подвергаться хешированию;
4. Система должна предоставлять как минимум два интерфейса - интерфейс пользователя - покупателя и интерфейс пользователя – магазина;
5. Для хранения данных о пользователях, товарах, заказах и их составов использовать СУБД SQLite.
6. Для разработки сервисов использовать язык программирования Python 3.4, используя фреймворк Flask.

### 1.3.4 Функциональные требования к системе интернет магазина

#### 1.3.4.1 Общие функциональные требования к системе:

Система должна представлять собой законченный программный комплекс. Внедряемая система должна обеспечивать выполнение следующих действий и функций:

1. С точки зрения покупателя

* Аутентификация пользователей
* Регистрация пользователей
* Просмотр списка всех товаров
* Просмотр списка всех фирм
* Просмотр товаров, которыми торгует определенная фирма
* Просмотр фирм, которые торгуют определенным товаром
* Просмотр данных о товаре
* Составление заказа
* Просмотр состава заказа

1. С точки зрения менеджера фирмы

* Аутентификация пользователей
* Регистрация пользователей
* Просмотр списка всех товаров
* Просмотр списка всех фирм
* Просмотр товаров, которыми торгует определенная фирма
* Просмотр фирм, которые торгуют определенным товаром
* Просмотр данных о товаре
* Просмотр заказов
* Редактирование статуса заказа для своей фирмы
* Редактирование предоставляемых товаров своей фирмы
* Просмотр состава заказа

#### 1.3.4.2 Входные данные

Входными параметрами для системы являются:

* + 1. Логин и пароль пользователя
    2. Код сессии
    3. Данные о товаре
  + Id товара
  + Марка производителя
  + Модель
  + Цвет
  + Сопроводительный текст

4. Данные о заказе

* + Id пользователя
  + Id магазина
  + Id товара

5. Данные о фирме

* Продаваемые товары
* Цена

#### 1.3.4.3 Выходные данные

На выходе система должна выдавать:

* + - 1. Список товаров
      2. Список фирм
      3. Описание фирм
* Название
* Продаваемый товар
  + - 1. Полное описание товара с ценой
* Наименование
* Марка
* Модель
* Цвет
* Фирма продавец
* Сопроводительный текст
* Цена
  + - 1. Данные о заказе
* Фирма продавец
* Товар
* Цена

### 1.3.5 Функциональные требования к системе фирмы

#### 1.3.5.1 Общие функциональные требования к системе:

Система должна представлять собой законченный программный комплекс. Внедряемая система должна обеспечивать выполнение следующих действий и функций:

1. С точки зрения менеджера фирмы:
   * + - Аутентификация пользователей
       - Регистрация пользователей
       - Просмотр товаров, которыми торгует определенная фирма
       - Просмотр данных о товаре
       - Просмотр заказов
       - Редактирование товара
       - Редактирование списка товаров
       - Редактирование статуса заказа
       - Добавление товара
       - Удаление товара
       - Просмотр состава заказа

#### 1.3.5.2 Входные данные

Входными параметрами для системы являются:

* + Информация о заказе
    1. Id пользователя
    2. Id магазина
    3. Id товара
  + Логин и пароль пользователя
  + Данные о товаре
    1. Наименование
    2. Марка
    3. Модель
    4. Цвет
    5. Сопроводительный текст
    6. Цена

#### 1.3.5.3 Выходные данные

На выходе система должна выдавать:

* + Список товаров
  + Описание товара
    1. Наименование
    2. Марка
    3. Модель
    4. Цвет
    5. Сопроводительный текст
    6. Цена
  + Данные о заказе
    1. Цена
    2. Покупатель
    3. Состав
    4. Состояние

### 1.3.6 Функциональные требования к системе удаленного сервера

#### 1.3.6.1 Общие функциональные требования к системе:

Система должна представлять собой законченный программный комплекс. Внедряемая система должна обеспечивать выполнение следующих действий и функций:

* + - 1. Резервное копирование базы данных
      2. Восстановление базы данных

#### 1.3.6.2 Входные данные

Входными параметрами для системы являются:

* + 1. Актуальная копия базы данных

#### 1.3.6.3 Выходные данные

На выходе система должна выдавать:

* + - 1. Резервная копия базы данных

## 1.4 Условия эксплуатации

Для обеспечения бесперебойного функционирования РСОИ на центральном сервере необходимо установить кондиционеры, поддерживающие следующие климатические условия:

1. Обеспечить температурный режим от 15 до 30 градусов Цельсия;
2. Относительная влажность воздуха - не более 80%;
3. Расположить компоненты сервера на жестких металлических стойках;
4. Оборудование должно быть заземлено.

## 1.5 Требования к составу и параметрам технических средств

Минимальные технические требования для центрального сервера:

1. 2-х ядерный процессор с тактовой частотой 2 ГГц;
2. ОЗУ 4 ГБ;
3. ПЗУ 300 ГБ для РСОИ;
4. Сетевая карта Ethernet стандарта 100BASE-T;
5. Монитор с разрешение экрана 1366х768 точек;
6. Клавиатура;
7. Мышь.

Общие требования к программному окружению:

1. Web-браузер Chrome 17.0.

## 1.6 Требования к надежности

Для повышения надёжности необходимо предусмотреть возможность установки дублирующего сервера для сервера, обеспечивающего работу пользователей через web-интерфейс, поскольку он является наиболее уязвимым и важным звеном в архитектуре системы.

Администратор баз данных должен обеспечить периодическое создание резервных копий базы данных (1 раз в сутки).

Для предотвращения сбоев в работе СУБД при сбое в подаче электропитания, необходимо обеспечить подключение серверов к устройствам бесперебойного электропитания, которые обеспечат не менее 30 минут автономной работы.

## 1.7 Требования к документации

Документация должна включать:

1. Руководство по развертыванию системы;
2. Руководство по использованию системы;
3. Документация к системе должна быть подготовлена на русском языке и представлена как на бумаге, так и в электронном виде.

Технический проект

1 Сценарий использования системы

Пользователям системы являются физические лица, которые хотят оформить заказ в электронном магазине. Таким образом, в системе выделяется две роли пользователей – «покупатель» и «продавец». Диаграмма прецедентов пользователя подсистемы электронного магазина изображена на рисунке 2.



Рисунок 2 Диаграмма прецедентов пользователя подсистемы электронного магазина.

1.1 Спецификация прецедента «Регистрация»

Цель прецедента использования: клиент регистрируется в системе, сообщая почтовый адрес, логин и пароль.

Оптимистический сценарий:

* Клиент вводит данные в окне веб-страницы
* Система запоминает данные в таблице базы данных
* Система перенаправляет пользователя на страницу пользователя

1.2 Спецификация прецедента «Вход в систему»

Цель прецедента использования: клиент входит в систему под своим аккаунтом.

Оптимистический сценарий:

* Клиент вводит данные в окно веб-страницы
* Система проверяет наличие данных о пользователе в базе данных
* Если данные существуют, система перенаправляет пользователя на страницу пользователя.

Альтернативный сценарий:

* Клиент вводит данные в окно веб-страницы
* Система проверяет наличие данных о пользователе в базе данных
* Если данные отсутствуют, система выводит сообщение о неверных введенных данных

1.3 Спецификация прецедента «Просмотр всех товаров»

Цель прецедента использования: клиент осуществляет просмотр списка всех товаров.

Оптимистический сценарий:

* Система проверяет наличие товаров в базе данных
* Если данные существуют, система показывает список товаров в окне веб-страницы

1.4 Спецификация прецедента «Просмотр всех фирм»

Цель прецедента использования: клиент осуществляет просмотр списка всех фирм.

Оптимистический сценарий:

* Система проверяет наличие товаров в базе данных
* Если данные существуют, система показывает список фирм в окне веб-страницы

1.5 Спецификация прецедента «Просмотр товара»

Цель прецедента использования: клиент осуществляет просмотр информации о товаре.

Оптимистический сценарий:

* Система проверяет наличие товара в базе данных
* Если данные существуют, система показывает данные о товаре в окне веб-страницы

1.6 Спецификация прецедента «Сделать заказ»

Цель прецедента использования: клиент осуществляет составление заказа.

Оптимистический сценарий:

* Система проверяет наличие товара в базе данных
* Если данные существуют, система перенаправляет пользователя на страницу подтверждения заказа
* После подтверждения заказа, заказ заносится в базу данных

Альтернативный сценарий:

* Система проверяет наличие товара в базе данных
* Если данные существуют, система перенаправляет пользователя на страницу подтверждения заказа
* Если подтверждения заказа не произошло, система перенаправляет пользователя на страницу списка всех товаров

1.7 Спецификация прецедента «Просмотр заказа»

Цель прецедента использования: клиент осуществляет просмотр заказа.

Оптимистический сценарий:

* Система проверяет наличие заказа в базе данных
* Если данные существуют, система показывает данные о заказе в окне веб-страницы

Альтернативный сценарий:

* Система проверяет наличие заказа в базе данных
* Если данные отсутствуют, система перенаправляет пользователя на страницу с описанием ошибки

1.8 Спецификация прецедента «Просмотреть все заказы»

Цель прецедента использования: клиент осуществляет просмотр списка всех заказов.

Оптимистический сценарий:

* Система проверяет наличие заказов в базе данных
* Если данные существуют, система показывает данные о товаре в окне веб-страницы

Альтернативный сценарий:

* Система проверяет наличие заказов в базе данных
* Если данные не существуют, система показывает сообщение о том, что ни одного заказа пока не сделано