

СОДЕРЖАНИЕ

Глоссарий	2
ВВЕДЕНИЕ	3
1 Аналитический раздел	4
1.1 Основания для разработки	4
1.2 Назначение разработки	4
1.3 Существующие аналоги	4
1.4 Описание системы	4
1.5 Общие требования к системе	5
1.6 Требования к функциональным характеристикам	5
1.7 Функциональные требования к portalу с точки зрения пользо- вателя	5
1.8 Требования к программной реализации	7
1.9 Топология системы	7
1.10 Общие требования к подсистемам	8
1.11 Функциональные требования к сервисам	8
1.12 Требования к надежности и к документации	14
2 Конструкторский раздел	15
2.1 Концептуальный дизайн	15
2.2 Сценарии функционирования системы	16
3 Технологический раздел	20
4 Исследовательский раздел	21
ЗАКЛЮЧЕНИЕ	22

Глоссарий

В данном техническом задании используются следующие обозначения:

1. Узел системы – региональный сервер, содержащий данные авторов и читателей указанного региона;
2. Валидация – проверка данных на соответствие заданным условиям и ограничениям;
3. REST – архитектурный стиль взаимодействия компонентов распределённого приложения в сети;
4. Медиана времени отклика – среднее время предоставления данных пользователю;
5. Латентность географического положения – увеличение времени отклика приложения, обуславливаемое географическим положением элементов системы или пользователя.
6. Аутентификация – процесс проверки подлинности пользователя или устройства.
7. Авторизация – это процесс проверки прав доступа.
8. OpenID Connect – это протокол аутентификации и авторизации, который строится на основе протокола OAuth 2.0.
9. Identity Provider – это сервис, который управляет аутентификацией и предоставляет информацию о пользователе в рамках системы аутентификации и авторизации.

ВВЕДЕНИЕ

В современном мире автосервисы становятся все более востребованными, и для их эффективного управления необходимо использовать специализированные порталы. Данный проект представляет собой техническое задание на разработку портала для автосервисов, который будет отслеживать состояние склада запчастей и выполненные сотрудниками работы. Этот портал позволит автосервисам улучшить управление запасами, повысить оперативность обслуживания клиентов и улучшить качество предоставляемых услуг. В данном техническом задании будут описаны основные функциональные требования к portalу, архитектура системы и основные интеграции с другими сервисами.

1 Аналитический раздел

1.1 Назначение разработки

Разрабатываемая система должна предоставлять возможность клиентам просматривать выполненные работы, наличие запчастей на складе, продавцам возможность просматривать и продавать запчасти, работникам просматривать выполненные работы, кладовщикам принимать заказы и разгружать их на складе, администраторам просматривать аналитику работы сервиса.

Для работников автосервиса должна быть предусмотрена возможность поиска запчастей на складе по артикулу, названию на русском и английском языках, а также возможность найти аналоги выбранной запчасти.

1.2 Существующие аналоги

Среди аналогов разрабатываемого проекта можно выделить Garage Management System и AutoSoft Online. Разрабатываемый проект должен обладать следующими преимуществами:

1. Поиск аналогов для выбранных запчастей;
2. Введение истории работ;
3. Просмотр наличия запчастей на складе клиентом.

1.3 Описание системы

Разрабатываемый сервис должен представлять собой распределенную систему для взаимодействия пользователей и работников с автосервисами.

Пользователь может выступать в качестве клиента, который может посмотреть свою историю работ и покупок, а также наличие запчастей на складе; в качестве продавца или кладовщика, которые взаимодействуют с деталями на складе и обрабатывают заказы; в качестве сотрудника, который может посмотреть информацию о выполненных работах или в качестве администратора, который может просматривать отчеты о работе автосервиса.

1.4 Общие требования к системе

1. Необходимо поддерживать возможность добавления нового узла во время работы системы без перезапуска;

2. Каждый узел должен автоматически восстанавливаться после сбоя;
3. Система должна автоматически выбирать наиболее подходящие серверы из доступных.

1.5 Требования к функциональным характеристикам

1. По результатам работы модуля сбора статистики медиана времени отклика системы на запросы пользователя на получение информации не должна превышать 3 секунд без учета латентности географического расположения узла;
2. По результатам работы модуля сбора статистики медиана времени отклика системы на запросы, добавляющие или изменяющие информацию на портале не должна превышать 5 секунд без учета латентности географического расположения узла.

1.6 Функциональные требования к portalу с точки зрения пользователя

Portal должен обеспечивать реализацию следующих функций;

1. Регистрация и авторизация пользователей с валидацией вводимых данных как через интерфейс приложения, так и через популярные социальные сети.
2. Аутентификация пользователей.
3. Ролевая модель пользователей. Выделяются следующие роли:
 - клиент;
 - кладовщик;
 - мастер;
 - продавец;
 - администратор.
4. Клиент имеет следующий набор функций:

- просмотр истории заказов;
- получение доступных на складе запчастей и списка возможных работ.

5. Кладовщик имеет следующий набор функций:

- получение информации расположении запчастей на складе;
- получение списка незакрытых инвойсов;
- положить деталь на склад;
- закрытие инвойса.

6. Продавец имеет следующий набор функций:

- создание инвойсов;
- создание клиентов;
- добавление машин клиентов;
- продажа запчастей со склада;
- продажа работ, выполненных мастером.

7. Мастер имеет следующий набор функций:

- поиск деталей на складе;
- просмотр истории выполненных.

8. Администратор имеет следующий набор функций:

- изменение ролей сотрудников;
- просмотр отчетов о работе сотрудников;
- поиск деталей на складе;
- добавление деталей и аналогов в базу.

1.7 Требования к программной реализации

1. Требуется использовать сервис-ориентированную архитектуру для реализации системы.

2. Система состоит из микросервисов. Каждый микросервис отвечает за свою область логики работы приложения.
3. Взаимодействие между сервисами осуществляется посредством HTTP запросов.
4. Данные сервисов должны храниться в базе данных. Каждый сервис взаимодействует только со своей схемой данных. Взаимодействие сервисов происходит по технологии REST.
5. При недоступности систем портала должна осуществляться деградация функциональности или выдача пользователю сообщения об ошибке.
6. Необходимо предусмотреть авторизацию пользователей, как через интерфейс приложения, так и через популярные социальные сети.
7. Для авторизации использовать OpenID Connect, в роли Identity Provider использовать стороннее решение.
8. Для запросов, выполняющих обновление данных на нескольких узлах распределенной системы, в случае недоступности одной из систем, необходимо выполнять полный откат транзакции.
9. Приложение должно поддерживать возможность горизонтального и вертикального масштабирования за счет увеличения количества функционирующих узлов и совершенствования технологий реализации компонентов и всей архитектуры системы.

1.8 Топология системы

Топология системы представлена на рисунке 1.1.

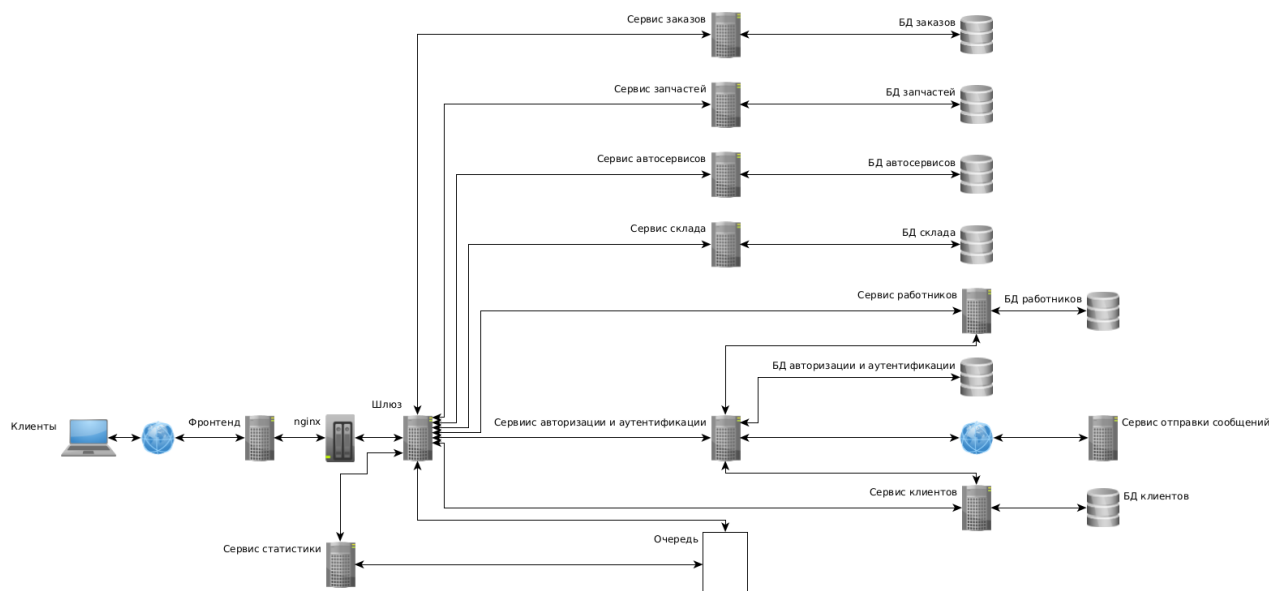


Рисунок 1.1 – Топология системы

1.9 Общие требования к подсистемам

1. Фронтенд должен принимать запросы от пользователя по протоколу HTTP и возвращать ответ в виде HTML страниц, файлов стилей и javascript.
2. nginx отвечает за балансировку трафика между несколькими серверами шлюза, если такие есть, возврат ответов на статические запросы – изображения, кеширование.
3. Взаимодействие между остальными сервисами осуществляется по протоколу HTTP, для передачи объектов используется формат JSON.

1.10 Функциональные требования к сервисам

Сервис работников.

Хранимая в базе данных сущность, ассоциированная с сервисом, должна иметь следующие обязательные поля:

- идентификатор;
- идентификатор автосервиса;
- имя;

- фамилия;
- должность;
- роль;

Сервис должен реализовывать следующий функционал:

- добавление работника;
- обновления работника;
- получение информации о работнике;
- получение всех работников;
- удаление работника.

Сервис клиентов.

Хранимая в базе данных сущность, ассоциированная с сервисом, должна иметь следующие обязательные поля:

- идентификатор;
- идентификатор автосервиса;
- имя;
- фамилия;
- номер телефона;
- флаг юридическое или физическое лицо;
- список машин.

Для описания машины используются следующие поля:

- vin номер;
- марка;
- модель;

- государственный регистрационный знак.

Сервис должен реализовывать следующий функционал:

- добавить машину;
- добавить клиента;
- получить информацию о клиенте;
- найти клиента по номеру телефона;
- обновить клиента;
- добавить клиенту машину.

Сервис авторизации и аутентификации

Сервис должен реализовывать следующий функционал:

- добавить логин и пароль работника;
- проверить логин и пароль работника;
- создать access и refresh токены работника;
- обновить access токен работника;
- отправить код по номеру телефона клиента;
- подтвердить номер телефона кодом и получить access и refresh токены клиента;
- обновить access токен клиента.

Сервис автосервисов

Хранимая в базе данных сущность, ассоциированная с сервисом должна иметь следующие поля:

- идентификатор;
- название;
- адрес;

- номер телефона.

Сервис должен реализовывать следующий функционал:

- добавить автосервис;
- получить все автосервисы;
- поиск автосервиса по имени;
- удалить автосервис.

Сервис склада.

Ассоциированная с складом сущность имеет следующие поля в базе данных:

- идентификатор склада;
- идентификатор автосервиса;
- название;
- адрес.

Ассоциированная с инвойсом сущность имеет следующие поля:

- идентификатор;
- артикул детали;
- идентификатор поставщика;
- цена;
- количество;
- статус.

Ассоциированная с местом на складе сущность имеет следующие поля:

- идентификатор места;
- идентификатор склада;

- название.

Ассоциированная с деталью на складе сущность имеет следующие поля:

- идентификатор;
- идентификатор места;
- идентификатор инвойса;
- флаг присутствия.

Ассоциированная с поставщиком сущность имеет следующие поля:

- идентификатор;
- название;
- контактные данные;
- тип (физическое или юридическое лицо).

Сервис должен реализовывать следующий функционал:

- добавить склад;
- удалить склад;
- добавить место на складе;
- удалить место на складе;
- создать инвойс;
- закрыть инвойс;
- взять деталь со склада;
- положить деталь на склад.

Сервис деталей.

Сущность ассоциированная с производителями имеет следующие поля в базе данных:

- идентификатор;
- имя;
- страна.

Сущность ассоциированная с деталями имеет следующие поля:

- артикул;
- идентификатор производителя;
- название на английском;
- название на русском.

Сущность ассоциированная с заменами имеет следующие поля:

- артикул запчасти;
- артикул запчасти.

Сервис должен предоставлять следующий функционал:

- получить список деталей;
- найти деталь по имени;
- добавить деталь;
- получить список производителей;
- получить список замен для выбранной запчасти;
- добавить замену.

Сервис заказов. Ассоциированная с заказом сущность имеет следующие поля в базе данных:

- идентификатор;
- имя;
- идентификатор клиента.

Ассоциированная с деталью в заказе сущность имеет следующие поля:

- идентификатор детали в заказе;
- идентификатор заказа;
- артикул детали;
- стоимость продажи;
- идентификатор продавца.

Ассоциированная с работой в заказе сущность имеет следующие поля в базе данных:

- идентификатор работы в заказе;
- идентификатор заказа;
- наименование;
- исполнитель;
- стоимость.

Сервис должен предоставлять следующий функционал:

- создать заказ;
- удалить заказ;
- добавить деталь в заказ;
- удалить деталь из заказа;
- добавить работу в заказ;
- удалить работу из заказа;
- получить все заказы клиента.

1.11 Требования к надежности и к документации

Система должна работать в соответствии с данным техническим заданием без перезапуска. Необходимо использовать «зеркалируемые серверы» для всех подсистем, которые будут держать нагрузку в случае сбоя до тех пор, пока основной сервер не восстановится.

2 Конструкторский раздел

2.1 Концептуальный дизайн

На рисунке 2.1 отображена контекстная диаграмма верхнего уровня, которая обеспечивает наиболее общее или абстрактное описание работы системы.

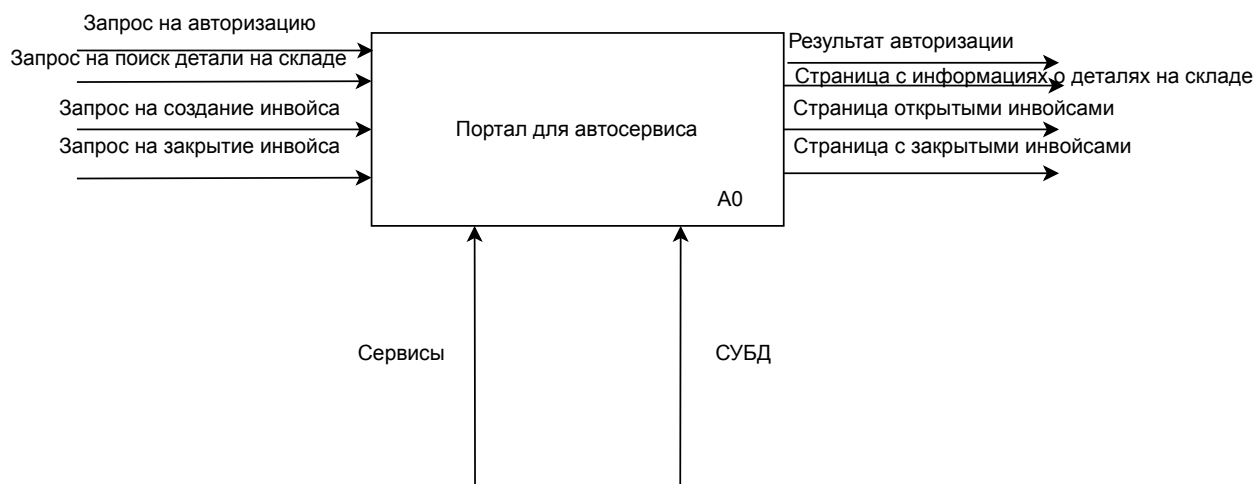


Рисунок 2.1 – Концептуальная модель в нотации IDEF0

На рисунке 2.2 изображена дочерняя диаграмма, которая определяет последовательность выполнения операций в системе при обработке запроса пользователя на создание инвойса.

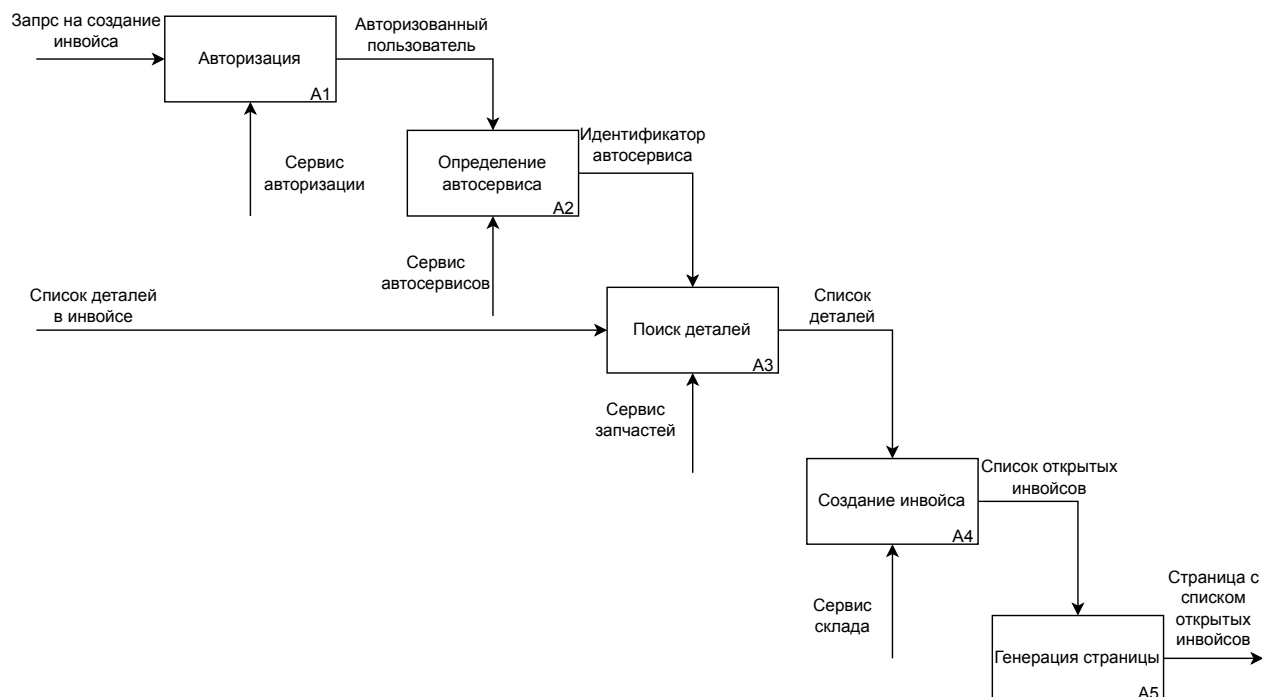


Рисунок 2.2 – Детализированная концептуальная модель в нотации IDEF0

2.2 Сценарии функционирования системы

Регистрация работника:

1. Работник нажимает на кнопку «Зарегистрироваться как работник» в интерфейсе приложения.
2. Работник перенаправляется на страницу регистрации, которая содержит поля для ввода.
3. Работник вводит данные в форму и для завершения регистрации нажимает на кнопку «Регистрация», тем самым подтверждая верность своих данных, а также согласие на их обработку и хранение.
4. Если пользователь с введенным для регистрации именем уже существует, то работник перенаправляется на страницу ошибки. При успешной регистрации он перенаправляется на страницу своего профиля в системе.

Регистрация клиента:

1. Клиент нажимает на кнопку «Зарегистрироваться как клиент» в интерфейсе приложения.

2. Клиент перенаправляется на страницу регистрации, которая содержит поля для ввода.
3. Клиент вводит данные в форму и для завершения регистрации нажимает на кнопку «Регистрация», тем самым подтверждая верность своих данных, а также согласие на их обработку и хранение.
4. Если пользователь с введенным для регистрации номером телефона уже существует, то клиент перенаправляется на страницу ошибки. При успешной регистрации он перенаправляется на страницу своего профиля в системе.

Авторизация работника:

1. Работник нажимает на кнопку «Войти как работник» в интерфейсе приложения.
2. Работник перенаправляется на страницу авторизации, которая содержит поля для логина и пароля.
3. Работник нажимает на кнопку войти.
4. При обнаружении ошибки в данных, работник перенаправляется на страницу ошибки, иначе он получает доступ к системе.

Авторизация клиента:

1. Клиент нажимает на кнопку «Войти как клиент» в интерфейсе приложения.
2. Клиент перенаправляется на страницу авторизации, которая содержит поле для ввода номера телефона.
3. Клиент нажимает на кнопку «Получить код».
4. Клиент перенаправляется на страницу, которая содержит форму для ввода кода.
5. Клиент вводит код из смс.
6. Клиент нажимает на кнопку войти.

7. Если введенный код не совпадает с отправленным, клиент перенаправляется на страницу ошибки, иначе он получает доступ к системе.

Создание инвойса

1. Авторизованный пользователь с ролью не ниже «продавец» нажимает на кнопку «Создать инвойс».
2. Пользователь перенаправляется на страницу с формой для создания инвойса.
3. Пользователь вводит артикулы деталей, количество, поставщика и цену поставки деталей каждого типа.
4. Пользователь нажимает на кнопку «Создать».
5. В случае успеха пользователь перенаправляется на страницу с открытыми инвойсами.

Закрытие инвойса

1. Авторизованный пользователь с ролью не ниже «кладовщик» выбирает один инвойс из открытых и нажимает на кнопку «Закрыть».
2. Пользователь перенаправляется на страницу с формой для закрытия.
3. Пользователь для каждой детали в инвойсе указывает место на складе, куда он ее положил.
4. Пользователь нажимает на кнопку «Закрыть».
5. В случае успеха пользователь перенаправляется на страницу с закрытыми инвойсами.

Создание заказа

1. Авторизованный пользователь с ролью не ниже «продавец» нажимает на кнопку «Создать заказ».
2. Пользователь перенаправляется на страницу с формой для создания заказа.

3. Пользователь выбирает клиента и вводит имя заказа.
4. Пользователь перенаправляется на страницу для редактирования заказа.
5. Пользователь добавляет детали и работы в заказ.
6. Пользователь нажимает на кнопку «Завершить редактирование».
7. Пользователь перенаправляется на страницу с заказами.

Получение статистики

1. Пользователь с ролью «администратор» нажимает на кнопку «Получить статистику».
2. Пользователь перенаправляется на страницу просмотра статистики о запросах.

3 Технологический раздел

4 Исследовательский раздел

ЗАКЛЮЧЕНИЕ