МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ

Федеральное государственное автономное образовательное учреждение высшего образования

«САНКТ-ПЕТЕРБУРГСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ   
АЭРОКОСМИЧЕСКОГО ПРИБОРОСТРОЕНИЯ»

КАФЕДРА № 43

ОТЧЕТ   
ЗАЩИЩЕН С ОЦЕНКОЙ

ПРЕПОДАВАТЕЛЬ

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| ассистент |  |  |  | А.Э. Зянчурин |
| должность, уч. степень, звание |  | подпись, дата |  | инициалы, фамилия |

|  |
| --- |
| ОТЧЕТ О ЛАБОРАТОРНОМ ЗАДАНИИ №1  Архитектура программного обеспечения |
| по дисциплине: " Основы программной инженерии" |
|  |
|  |

РАБОТУ ВЫПОЛНИЛ

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| СТУДЕНТ ГР. | 4134 |  |  |  | Самарин Д. В. |
|  |  |  | подпись, дата |  | инициалы, фамилия |

Санкт-Петербург

2023

# Описание предметной области.

Общие положения:

Компания, занимающаяся продажей компьютеров и комплектующих, предоставляет клиентам возможность приобрести высококачественные компьютерные системы, а также комплектующие для собственной сборки.

Проблематика:

Основная проблема, стоящая перед фирмой по сборке и продаже компьютеров, заключается в обеспечении эффективного и автоматизированного управления всеми аспектами бизнеса. Это включает в себя решение проблем с учетом инвентаря, отслеживание заказов, управление поставками комплектующих, а также обеспечение безопасности и надежности информационной системы.

Кроме того, компания сталкивается с вызовами выбора оптимальных технологий и программных решений для максимизации эффективности производственных процессов и обеспечения удовлетворения потребностей клиентов. Важными аспектами являются также разработка удобного интерфейса для пользователей и обеспечение интеграции различных модулей программного обеспечения для синхронизации работы всех подразделений компании.

# Цель и постановка задачи разработки программного обеспечения автоматизированной/информационной системы.

Цель разработки программного обеспечения автоматизированной/информационной системы состоит в создании эффективной и удобной платформы для предоставления компьютеров и комплектующих клиентам. Система должна обеспечивать быстрое и удобное нахождение наиболее подходящих под запросы клиента комплектующих.

Постановка задачи разработки программного обеспечения автоматизированной/информационной системы включает в себя следующие этапы:

1. Анализ требований: определение функциональных и нефункциональных требований к системе, а также оценка потребностей пользователей.

2. Проектирование архитектуры: разработка общей структуры системы, выбор технологий и оборудования, создание базы данных и разработка интерфейса пользователя.

3. Разработка: написание кода и тестирование системы на соответствие требованиям и стандартам безопасности.

4. Тестирование: проверка работоспособности системы, выявление ошибок и устранение их.

# Словарь данных предметной области.

1. Клиент — физическое или юридическое лицо, заключившее договор с фирмой по сборке и продаже компьютеров на приобретение продукции или услуг.
2. Сборка компьютера — процесс создания компьютерных систем путем объединения различных комплектующих, таких как процессоры, видеокарты, память и другие, в функциональное устройство.
3. Программное обеспечение — набор программных продуктов, разработанных для эффективной автоматизации бизнес-процессов компании, включая учет инвентаря, обработку заказов, управление поставками и другие аспекты.
4. Инвентарь — полный перечень комплектующих и готовых систем, находящихся в наличии у компании, включая их характеристики и количество.
5. Заказ — запрос клиента на приобретение определенного товара или услуги, который затем обрабатывается системой управления заказами.
6. Безопасность информационной системы — комплекс мероприятий и технологий, направленных на защиту конфиденциальности и целостности данных, а также обеспечение доступа к ним только авторизованным пользователям.
7. Интерфейс пользователя — средство взаимодействия клиента с программным обеспечением, предоставляющее удобный и интуитивно понятный способ взаимодействия с функционалом системы.

# Структура данных

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Элемент данных | Описание | Структура или тип данных | Количество | Значение |
| Заявка приобретение пк или комплектующих | " Заявка приобретение пк или комплектующих" представляет собой форму, в которой пользователи выбирают необходимые им комплектующие. Эта форма содержит информацию о пользователе, такую как имя, фамилия, адрес электронной почты и номер телефона. | ФИО клиента,  Дата заявки  Адрес доставки  Номер для связи | 1 |  |
| Конфигурация | “Конфигурация”  Представляет собой набор комплектующих, их стоимость, информации о товаре и гарантии, предлагаемых магазином | Цена Дата гарантии |  |  |
| Список конфигураций готового пк | Коллекция различных конфигураций в различных ценовых диапазонах для готовых пк | Список конфигураций |  |  |
| Личный кабинет | "Личный кабинет" представляет собой онлайн-платформу, которая позволяет пользователям управлять своими аккаунтами и получать доступ к различным сервисам и функциям. | ID История заказов Дата получения товара |  |  |
| База данных пользователей | "База данных пользователей" представляет собой базу данных, которая содержит информацию о всех зарегистрированных пользователях. Она содержит информацию о имени пользователя, адресе электронной почты, номере телефона и другую информацию, необходимую для идентификации пользователя. | ФИО Адрес  MAC адрес  Тариф  Номер телефона |  |  |

# Нежелательные эффекты в предметной области.

1. Неэффективное управление инвентаризацией: Проблемы с точностью отслеживания остатков товаров, неправильное управление запасами и несвоевременное обновление информации о наличии комплектующих может привести к сложностям в процессе сборки и поставок.
2. Недостаточная защита данных о клиентах: Неправильная реализация мер безопасности может привести к утечкам личной информации клиентов, что создаст репутационные и юридические проблемы для компании.
3. Сложности в обновлении программного обеспечения: Недостатки в структуре программы могут затруднить процесс обновления системы, что повлияет на возможности внедрения новых функций, адаптацию к изменяющимся рыночным требованиям и обеспечение безопасности.
4. Неудовлетворительное пользовательское взаимодействие: Непривлекательный интерфейс или неудобные функции могут снизить удовлетворенность клиентов и сотрудников, что отразится на общей репутации компании.

# Диаграмма вариантов использования.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | Заказ | Клиент | Магазин | БД клиентов |
| Создать заказ | С | С | RD | UD |
| Выполнить Заказ |  | R | U | UD |
| Управление БД клиентов |  |  | U | UD |
| Доставка заказа |  | C | R | U |

# Дерево функций программного обеспечения автоматизированной/информационной системы.

1. Функция "Просмотр готовых конфигураций пк"

2. Функция "Просмотр комплектующих"

3. Функция "Выбор нужной конфигурации пк"

4. Функция "Возможность собрать кастомный пк"

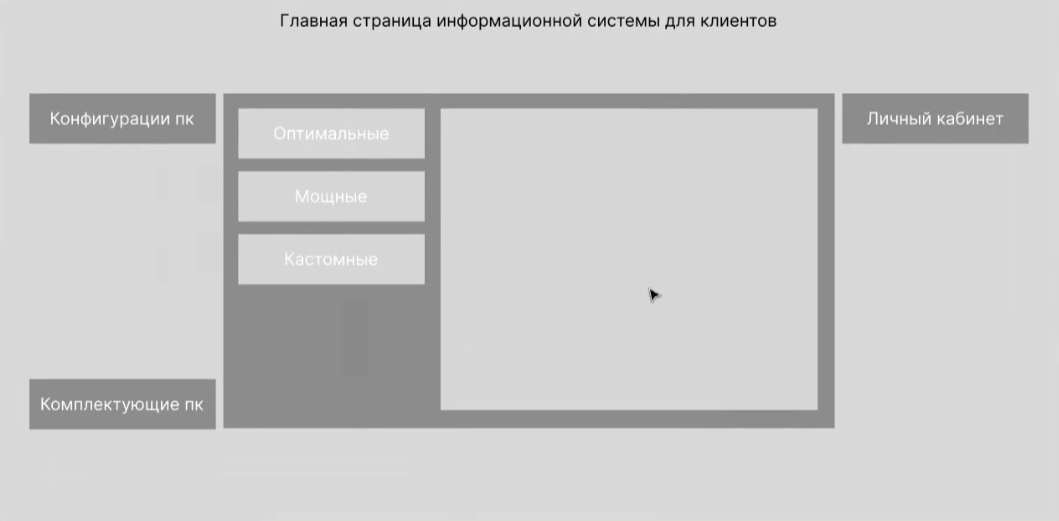
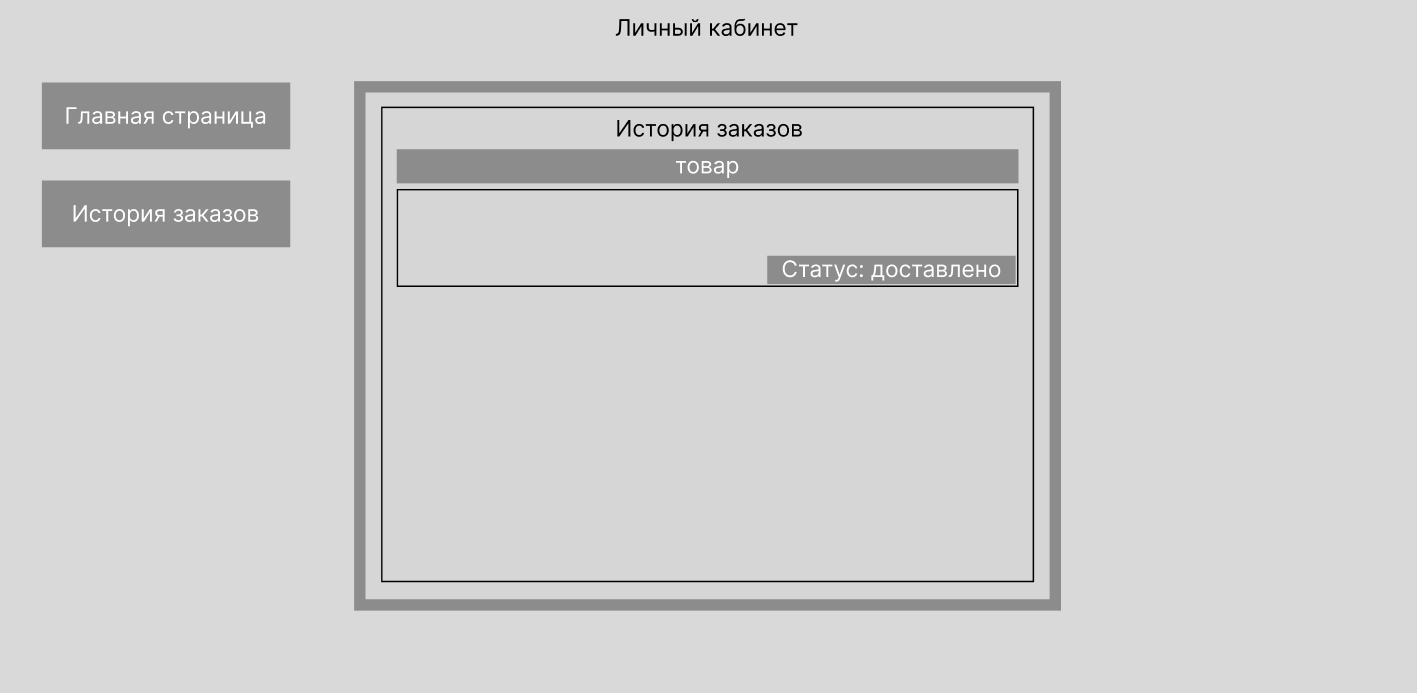
5. Функция "Просмотр личного кабинета"

6. Функция "Просмотр статуса заявки"

7. Функция " Указать ФИО клиента"

8. Функция " Указать Адрес клиента "

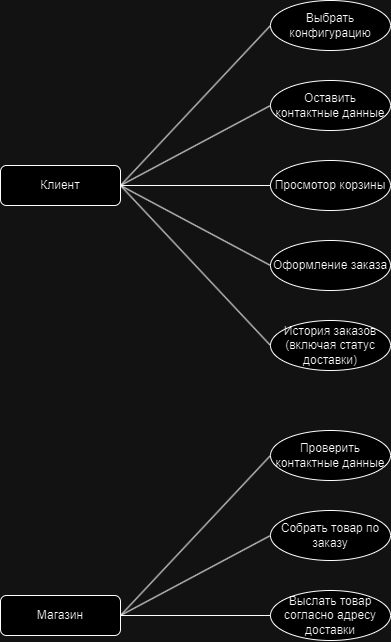
9. Функция " Указать контактный номер телефона клиента "

# Заключение и выводы.

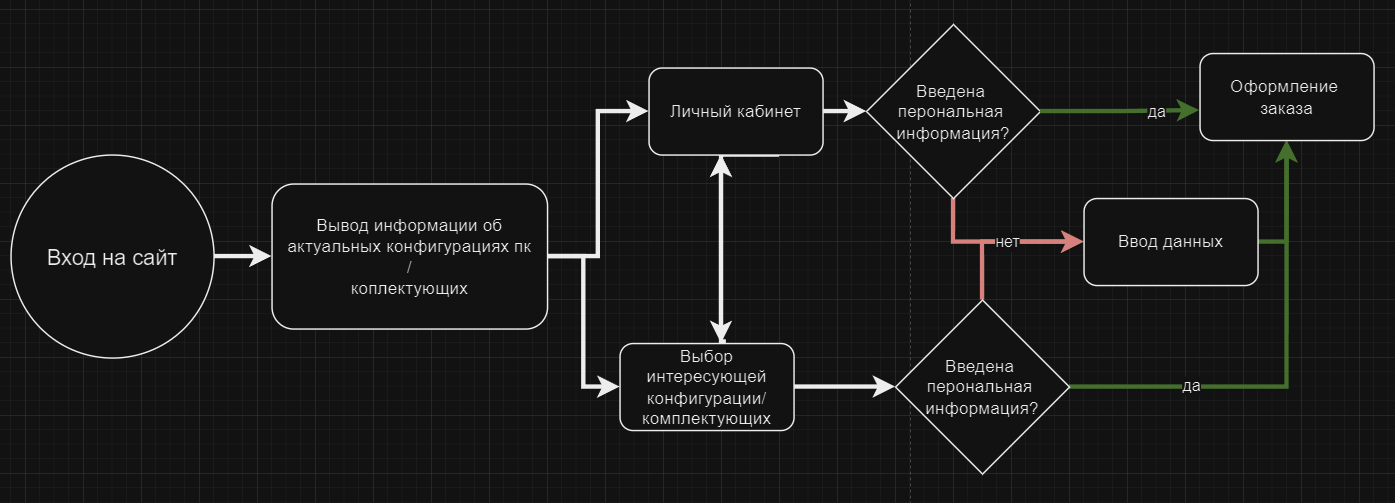
В завершение можно отметить, что разработка программного обеспечения для автоматизированной/информационной системы фирмы по сборке и продаже компьютеров и комплектующих играет ключевую роль в оптимизации бизнес-процессов и улучшении общего опыта как для компании, так и для её клиентов.

**Диаграмма прецедентов**

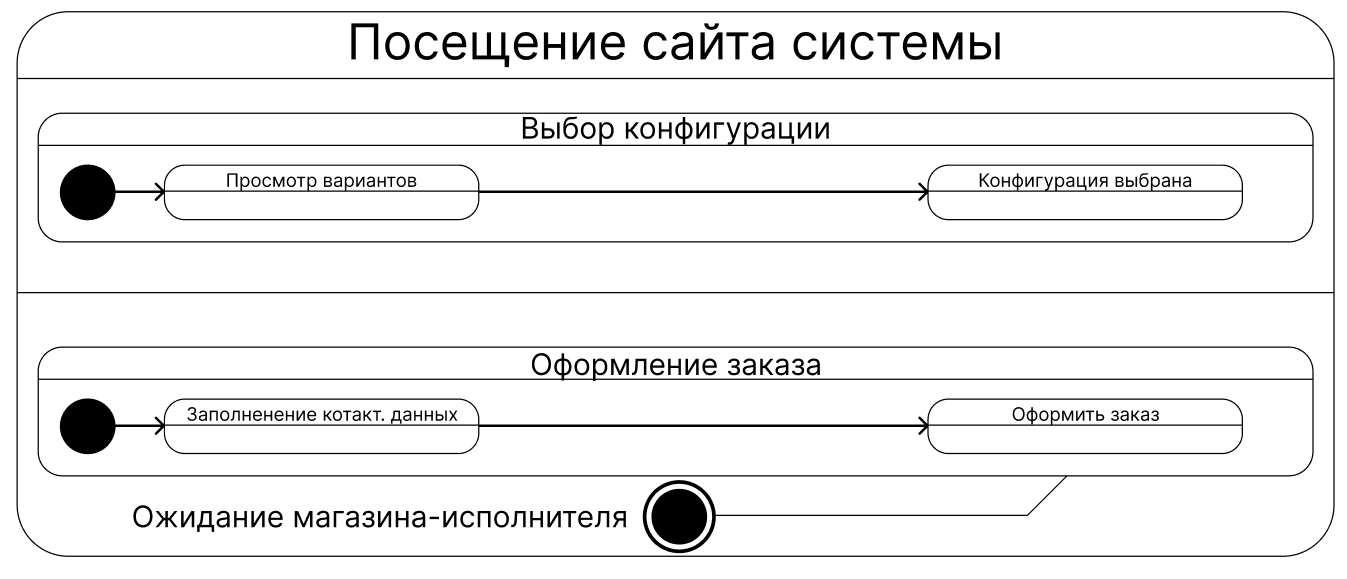


# Диаграмма объектов





# Диаграмма состояний





# Заключение и выводы:

Были выявлены связи между системой, магазином и клиентом

# Определение бизнес-требований

Для определения бизнес-требований необходимо провести анализ рынка и конкурентов, изучить потребности пользователей, определить ключевые показатели эффективности и принять во внимание ограничения бюджета и времени.

В результате анализа бизнес-требований можно сформулировать следующие требования к системе:

1. Функциональные требования:

**Подключение к сети связи:**

Обеспечение стабильного и надежного подключения к интернету для эффективного взаимодействия с системой.

**Передача данных:**

Реализация механизмов передачи данных между пользователями, клиентами и системой для обеспечения точной и своевременной информации.

**Безопасность и конфиденциальность данных:**

Гарантирование высокого уровня защиты данных клиентов и компании, включая шифрование информации и меры по предотвращению несанкционированного доступа.

**Управление услугами и ресурсами:**

Разработка функционала для эффективного управления услугами, контроля ресурсов и оптимизации бизнес-процессов.

2. Нефункциональные требования:

**Качество связи:**

Обеспечение стабильного качества связи для бесперебойного взаимодействия пользователей с системой.

**Надежность системы:**

Гарантирование высокой степени надежности и минимизации сбоев для бесперебойного функционирования.

**Простота использования:**

Разработка интуитивно понятного интерфейса, минимизирующего количество шагов для пользователей при оставлении заявок.

**Экономичность:**

Оптимизация затрат на разработку, поддержку и обслуживание системы.

Требуется разработать WEB приложение, которое будет показывать список конфигураций пк, иметь личный кабинет пользователя и форму для оформления заказа. Со стороны исполнителя требуется своевременно откликаться на заявку и связываться с клиентом по номеру телефона, который тот указал в заявке. От системы управления БД требуется наличие информации о поступивших заявках, статусе их выполнения, содержание актуально информации о клиенте и заказе, ожидаемой дате доставки. Интерфейс должен быть близок к минимальному количеству “Нажатий” для оформления заказа, чтобы упростить процесс покупки.

# Определение функциональных требований

Функциональные требования — это те функции и возможности, которые должны быть реализованы в программном обеспечении автоматизированной/информационной системе. Они описывают, какие действия пользователи могут выполнять с помощью системы, какие данные они могут получать и какие ресурсы они могут использовать.

Функциональные требования могут включать в себя:

1. Оформление заказа:

— Выбор конфигурации готового пк;

— Выбор комплектующих для пк.

2. Оформление заказа:

— Ввод заказчиком персональных данных (ФИО);

— Ввод заказчиком контактных данных(номер телефона и адрес);

3. Безопасность и конфиденциальность данных:

— Шифрование данных;

— Аутентификация пользователей;

Система предлагает на выбор некоторое количество готовых конфигураций пк и, при необходимости, отдельные комплектующие. Присутствует личный кабинет для оформления заказа. Субд клиентов должна ожидать поступления информации от исполнителя и обновлять актуальные данные.

# Определение нефункциональных требований

Нефункциональные требования — это те характеристики системы, которые не связаны с ее функциями, но имеют большое значение для ее успешной работы. Они описывают, каким образом система должна работать, чтобы удовлетворить потребности пользователей и достичь поставленных целей.

1. Скорость передачи данных:

— Высокая скорость загрузки файлов;

— Быстрый ответ на запросы пользователей;

— Минимальное время ответа на запросы пользователей.

2. Надежность системы:

— Минимальное количество сбоев в работе системы;

— Высокая степень надежности системы;

— Высокая степень безопасности системы.

3. Простота использования:

— Простой и интуитивно понятный интерфейс пользователя;

— Минимальное количество ошибок и недоразумений при использовании системы;

— Легкость управления системой.

4. Экономичность:

— Минимальные затраты на оборудование и обслуживание;

— Высокая степень эффективности использования ресурсов;

**Внешние интерфейсы**

Общение между частями системы происходит с помощью JSON и POST GET запросов HTTP.

**Атрибуты качества**

Для описания конфигураций пк предполагается использование таких слов как “Оптимальная”, “Мощная” и “Кастомная”.

# Информационные требования, предъявляемые к пользовательским интерфейсам

Информационные требования, предъявляемые к пользовательским интерфейсам, могут включать в себя:

1. Простоту и интуитивность: пользовательский интерфейс должен быть простым и интуитивно понятным, чтобы пользователи могли быстро освоить систему и легко находить нужную информацию.

2. Наглядность и читаемость: пользовательский интерфейс должен быть наглядным и читаемым, чтобы пользователи могли быстро ориентироваться в системе и легко находить нужную информацию.

3. Структурированность и логичность: пользовательский интерфейс должен быть структурированным и логичным, чтобы пользователи могли быстро находить нужную информацию и выполнять необходимые действия.

4. Гибкость и адаптивность: пользовательский интерфейс должен быть гибким и адаптивным, чтобы он мог легко адаптироваться к различным устройствам и операционным системам.

5. Стиль и дизайн: пользовательский интерфейс должен иметь привлекательный стиль и дизайн, который будет соответствовать бренду компании и ожиданиям пользователей.

# Функциональные требования, предъявляемые к пользовательским интерфейсам

1. Возможность поиска и получения информации: пользовательский интерфейс должен предоставлять возможность быстрого поиска и получения необходимой информации.

2. Возможность управления системой: пользовательский интерфейс должен предоставлять возможность управления системой, например, изменение составляющих для конфигураций пк.

3. Возможность интеграции с другими системами: пользовательский интерфейс должен предоставлять возможность интеграции с другими системами, например электронной почтой и т.д.

# Заключение и выводы:

В заключении можно сказать, что разработка программного обеспечения автоматизированной/информационной системы требует тщательного анализа требований и определения функциональных и нефункциональных требований. Важно учитывать потребности пользователей, конкурентную среду и ограничения бюджета и времени. Правильно разработанная система должна обеспечивать высокое качество, высокую скорость отклика пользовательского интерфейса, безопасность и конфиденциальность данных, простоту использования и экономичность.

Определение состава структурных элементов архитектуры программного обеспечения автоматизированной/информационной системы, их назначения и интерфейсов

|  |
| --- |
| Обработчик заяказов |
| **Назначение модуля:** Получает, изменяет и удаляет заказы. |
| **Варианты использование через состав методов модуля на псевдокоде:**  Изменить заказ (что, на что)  Подтвердить заказ()  Удалить заказ() |
| **Структуры и типы данных модуля:**  номер заявки: int  id конфигурации: int  id комплектующего: int  статус заказа: bool  дата заказа: str |
| **Декомпозиция:** связан с пользователем |
| **Класс или обобщение:** |

|  |
| --- |
| Клиент |
| **Назначение модуля:** Получение/редактирование данных о пользователе (ФИО, номер телефона, адрес). |
| **Варианты использование через состав методов модуля на псевдокоде:**  Изменить данные(что, на что)  Получить данные() |
| **Структуры и типы данных модуля:** |
| **Декомпозиция:** связан с Заказом |
| **Класс или обобщение:** |

|  |
| --- |
| История заказов (GET-запрос на сервер) |
| **Назначение модуля:** Позволяет получать историю заказов. |
| **Варианты использование через состав методов модуля на псевдокоде:**  Получить историю заказов() |
| **Структуры и типы данных модуля:**  История (номер конфигурации пк/комплектующих, набор дат и итоговой суммы): массив |
| **Декомпозиция:** связан с Клиентом |
| **Класс или обобщение:** |

|  |
| --- |
| Администратор |
| **Назначение модуля:** Редактирование данных о заказах, просмотр всей истории заказов и статусов заказов. |
| **Варианты использование через состав методов модуля на псевдокоде:**  Изменить заказ(id заявки)  Просмотр данных пользователей() |
| **Структуры и типы данных модуля:** |
| **Декомпозиция:** связан с Событием, Клиентом и Заказом |
| **Класс или обобщение:** |

|  |
| --- |
| Заказ(POST-запрос на сервер) |
| **Назначение модуля:** Хранение данных о адресе, клиенте и тд. |
| **Варианты использование через состав методов модуля на псевдокоде:**  Добавить заказ(фио, номер, адрес, id товара )  Получить данные о завке() |
| **Структуры и типы данных модуля:** |
| **Декомпозиция:** связан с Клиентом и Обработчиком заявок |
| **Класс или обобщение:** |

# Определение варианта архитектуры и ее разработка

Вариант архитектуры: на потоках данных

**Ограничения**

Для реализации использовать:

-FastAPI framework

-ЯП Python

-Html

-CSS

-JavaScript

Для реализации СУБД клиентов:

-postgresql

Для реализации исполнения подключения:

-технические навыки администрирования локальных сетей исполнителя

# Заключение и выводы:

В результате разработки документа были определены состав структурных элементов программного обеспечения автоматизированной информационной системы, их назначения и интерфейсы. Был определен вариант архитектуры и выполнена ее концептуальная разработка.