



МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ
Федеральное государственное бюджетное образовательное учреждение высшего
образования
"МИРЭА - Российский технологический университет"
РТУ МИРЭА

Институт информационных технологий (ИТ)
Кафедра практической и прикладной информатики (ППИ)

ИТОГОВЫЙ ОТЧЕТ ПО ПРАКТИЧЕСКИМ РАБОТАМ
по дисциплине
«Анализ и концептуальное моделирование систем»

Выполнил студент группы ИКБО-33-22

Шило Ю.С.

Принял старший преподаватель

Свищёв А.В.

Лабораторная работа выполнена

«__»_____202__ г.

(подпись студента)

«Зачтено»

«__»_____202__ г.

(подпись руководителя)

Москва 2024

ОГЛАВЛЕНИЕ

Отчет по практической работе № 1	3
Отчет по практической работе № 2	7
Отчет по практической работе № 3	10
Отчет по практической работе № 4	11
Отчет по практической работе № 5	15
Отчет по практической работе № 6	21
Отчет по практической работе № 7	26
Доклад	28
Заключение	30

Отчет по практической работе № 1

Тема: «описание функционала системы.»

Цель работы: изучить структуру и функционал организации работы платных курсов в образовательном учреждении

Задачи: необходимо детально описать функционал системы в соответствии с индивидуальным вариантом учебного проекта.

ОПИСАНИЕ ОБЪЕКТА АВТОМАТИЗАЦИИ

Для проведения анализа, я выбрал сайт своей школы, а именно «nf-sch5.edumsko.ru». В процессе внимательного изучения, я обратил внимание на несколько важных особенностей этого сайта. Во-первых, на нем предоставлена возможность подписаться на новости и получать информацию о событиях и активностях, связанных с учебным заведением. Кроме того, на сайте имеется ссылка на одну из социальных сетей. Еще одним положительным моментом является наличие версии, которая адаптирована для людей со слабым зрением.

Однако, при анализе сайта также были выявлены некоторые недостатки. Во-первых, отсутствует прямой доступ к перечню платных услуг, предоставляемых учебным заведением. Это может затруднять доступ к информации о дополнительных услугах и стоимости, что является недостатком для потенциальных клиентов. Кроме того, на сайте отсутствует форма для подачи заявки в онлайн формате. Другим недостатком является отсутствие чат-бота, который мог бы обеспечить быструю связь и решение некоторых вопросов. Кроме того, сайт не обеспечивает удобного доступа для просмотра информации о том, что представляет собой оплачиваемая услуга. Это может вызывать затруднения у пользователей, которые хотели бы получить более подробную информацию о платных услугах.

ОПИСАНИЕ ОСНОВНЫХ ФУНКЦИЙ СИСТЕМЫ

Таблица 1 – Основные функции для добавления

Наименование	Краткое описание
--------------	------------------

Информация об предоставляемых услугах	К каждой услуге добавить подробное описание время занятий и продолжительность курса, также добавить имена возможных преподавателей.
Информация о количестве свободных мест на курсе	Добавить информацию о количестве свободных мест свободных для записи.
Онлайн-помощник или чат бот	Добавить функцию для быстрой и легкой связи с учебным учреждением.
Онлайн форма для записи на занятия	Добавить форму для легкой записи на курсы в данном учебном учреждении
Быстрая ссылка на услуги	Добавить ссылку или меню для быстрого доступа к платным услугам данной школы
Личный кабинет пользователя	Добавить возможность создавать профиль для пользователя для удобного отслеживания состояния своих курсов.

На основе проведенного анализа сделаем вывод об основных функциях, которые следует добавить:

- Для каждой услуги необходимо предоставить подробное описание, указать время занятий, продолжительность курса и имена возможных преподавателей.
- Указать количество доступных свободных мест для записи.
- Добавить функцию для удобной и быстрой связи с учебным учреждением.
- Создать форму, которая позволит легко записаться на курсы в данном учебном учреждении.

- Включить ссылку или меню для быстрого доступа к платным услугам данной школы.

- Предоставить возможность пользователям создавать профиль для удобного отслеживания состояния своих курсов.

ОПИСАНИЕ ОЖИДАЕМЫХ РЕЗУЛЬТАТОВ

РЕАЛИЗАЦИЯ СИСТЕМЫ

Ожидаемым результатом после выполнения данных пунктов можно отнести:

- Расширение числа клиентов
- Повышение доходов
- Упрощение процесса поиска товаров
- Создание привлекательной среды для пользователей
- Обеспечение удержания потенциальных покупателей на сайте
- Сокращение времени, затрачиваемого на поиск нужной услуги

ВЫВОД

В результате выполнения данной практической работы был произведен анализ моделей с последующим выявлением их особенностей, функциональности и недостатков.

На основе проведенного анализа было выявлено следующие, а именно для сайта “”, можно добавить следующие:

- Информация об предоставляемых услугах
- Информация о количестве свободных мест на курсе
- Онлайн-помощник или чат бот
- Онлайн форма для записи на занятия
- Быстрая ссылка на услуги
- Личный кабинет пользователя

Ожидаемые результаты реализации моделируемой системы включают в себя улучшенную функциональность сайта и увеличение количества возможных клиентов.

Спроектировали диаграмму вариантов использования, определили активные субъекты и их взаимодействие с системой, добавив соответствующие варианты использования и актёров.

Расставили связи между элементами диаграммы, обеспечивая понимание взаимодействия пользователей с системой и основных функций системы.

Отчет по практической работе № 2

Тема: «описание функций системы через диаграмму вариантов использования»

Цель работы

Изучить основные элементы и правила построения диаграммы вариантов использования.

Задание 1

Построить диаграмму вариантов использования по следующему описанию: «Клиент банка может пополнить счет, в случае отсутствия счета предварительно открыв его, или снять деньги со счета, с возможностью его закрытия. В каждом из описанных действий участвует операционист банка и кассир.»

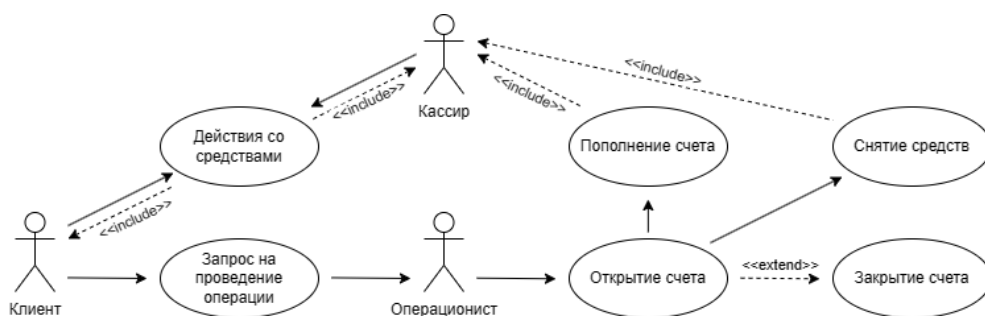


Рисунок 1 – UML диаграмма по описанию

Таблица 1 – Описание взаимодействия актеров и вариантов использования

Актёр/ВИ	Тип связи	Вариант использования
Клиент	Направленная ассоциация	Запрос на проведение операции
Клиент	Направленная ассоциация	Действия со средствами
Действия со средствами	Включение	Клиент
Действия со средствами	Включение	Кассир
Запрос на проведение операции	Направленная ассоциация	Операционист

Операционист	Направленная ассоциация	Открытие счета
Кассир	Направленная ассоциация	Действия со средствами
Открытие счета	Направленная ассоциация	Пополнение счета
Открытие счета	Расширение	Заккрытие счета
Пополнение счета	Включение	Кассир
Снятие средств	Включение	Кассир

Задание 2

Описать спецификацию функций рассматриваемой системы с учетом индивидуального варианта учебного проекта.

Перед построением диаграммы необходимо задокументировать потоки событий в системе.

Поток событий – процесс обработки данных, реализуемый в рамках одного или нескольких вариантов использования. Описание потока включает информацию о том, какие обязанности возлагаются на актеров, а какие на систему.

- 1) Студент (актер) запрашивает у Администрации (актер) доступ на курс. Студент отправляет заявку, Администрация должна их обработать.
- 2) Администрация (актер) создает профиль обучающегося
- 3) Студент (актер) отправляет выполнение задания, Преподаватель (актер) проверяет их.
- 4) Студент (актер) вместе с Преподаватель (актер) проходит курс.
- 5) После прохождения курса Администрация (актер) выдает сертификат об успешном прохождении курса.
- 6) Администрация (актер) выделяет бюджет.
- 7) Бюджет тратиться на оплату труда Преподаватель (актер).

- 8) Студент (актер) оплачивает услуги, предоставляемые школой пополняя тем самым бюджет организации.
- 9) Администрация (актер) создает профиль студента.
- 10) Преподаватель (актер) и Администрация (актер) составляют учебную программу, которую можно будет изменять.

Задание 3

Изобразить спецификацию функций системы, описанной в п.2 через диаграмму вариантов использования.

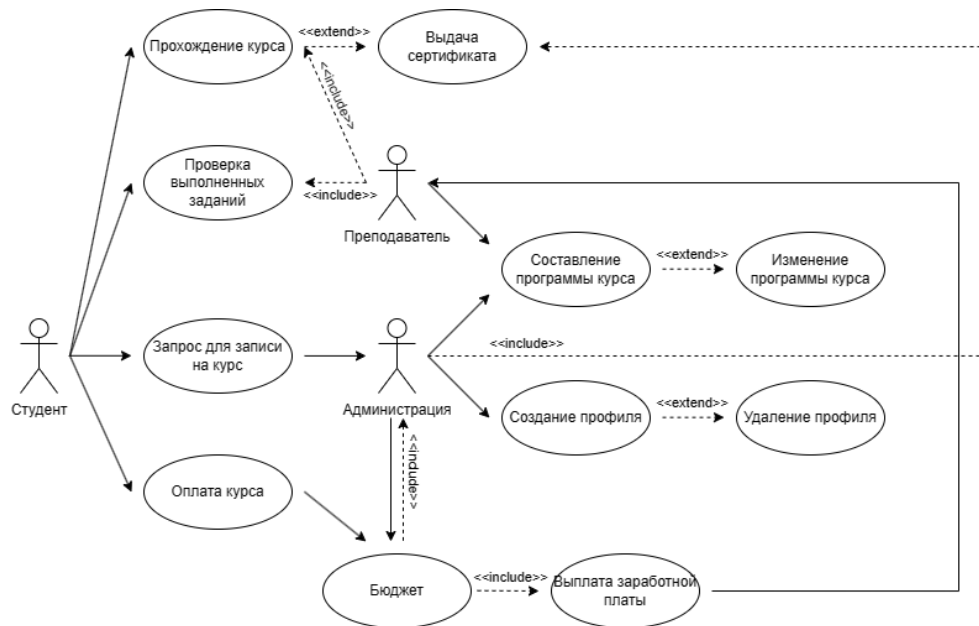


Рисунок 2 – UML диаграмма по варианту

Отчет по практической работе № 3
Тема: «построение uml – модели системы. Диаграмма классов анализа.»»

ЦЕЛЬ РАБОТЫ

Изучить структуру иерархии классов системы.

ЗАДАЧИ

Научиться выстраивать структуру основных элементов диаграммы классов анализа с определением видов классов и типов отношений.

ПРОГРАМНОЕ ОБЕСПЕЧЕНИЕ

Visual Paradigm, Draw.io, Rational Rose.

ВАРИАНТ

№29 – Моделирование работы сервисного центра.

ПОРЯДОК ВЫПОЛНЕНИЯ РАБОТЫ

Построенная диаграмму классов анализа рассматриваемой системы с учетом индивидуального варианта предоставлена на рисунке 1.

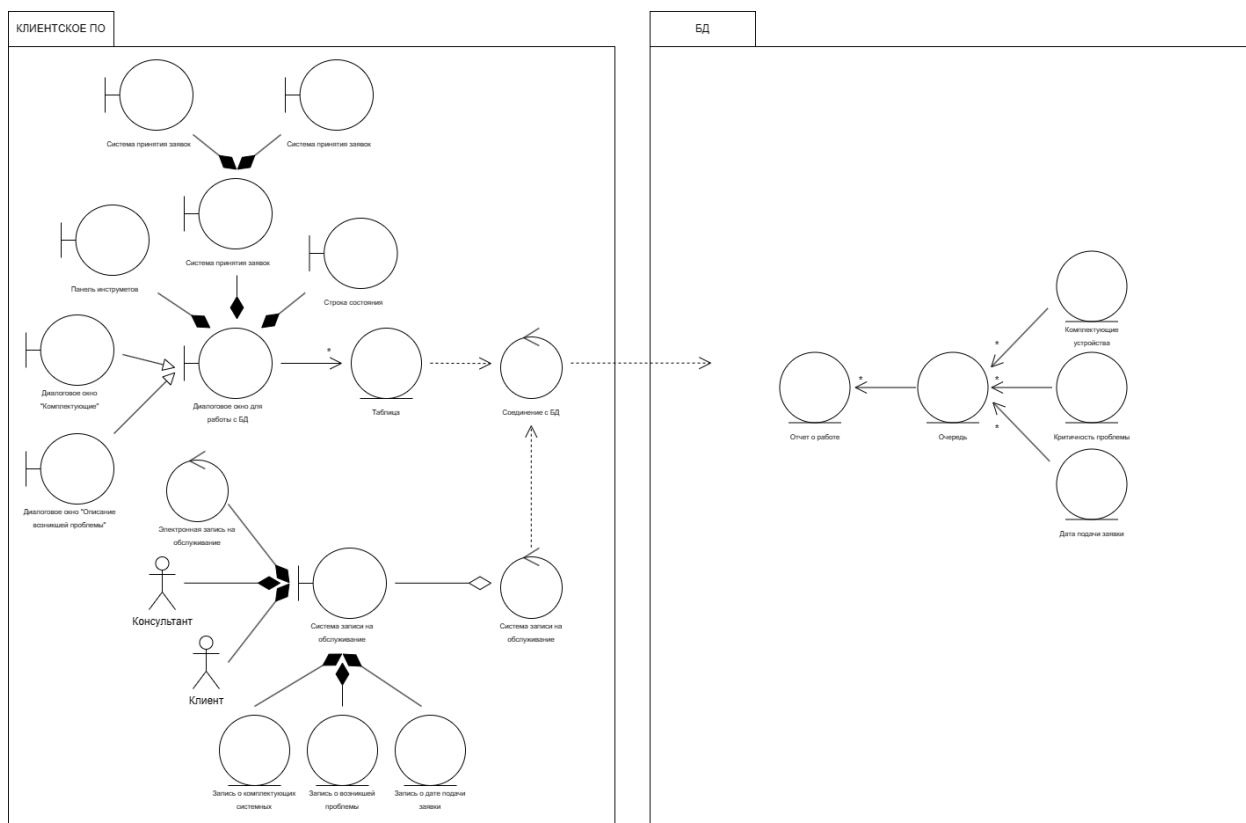


Рисунок 1 – Диаграмма классов анализа системы

Отчет по практической работе № 4
Тема: «Описание функций системы через диаграмму вариантов использования
использования.»

Цель работы

Изучить структуру и функционал рассматриваемой информационной системы, освоить правила построения диаграммы вариантов использования.

Вариант

29 => Моделирование работы сервисного центра.

Решение

Задание 1

Построить диаграмму последовательности по описанию приведенного варианта использования: «Студент хочет записаться на некий семинар, предлагаемый в рамках некоторого учебного курса. С этой целью проводится проверка подготовленности студента, для чего запрашивается список (история) семинаров курса, уже пройденных студентом (перейти к следующему семинару можно, лишь проработав материал предыдущих занятий). После получения истории семинаров объект класса "Слушатель" получает статус подготовленности, на основе которой студенту сообщается результат (статус) его попытки записи на семинар».

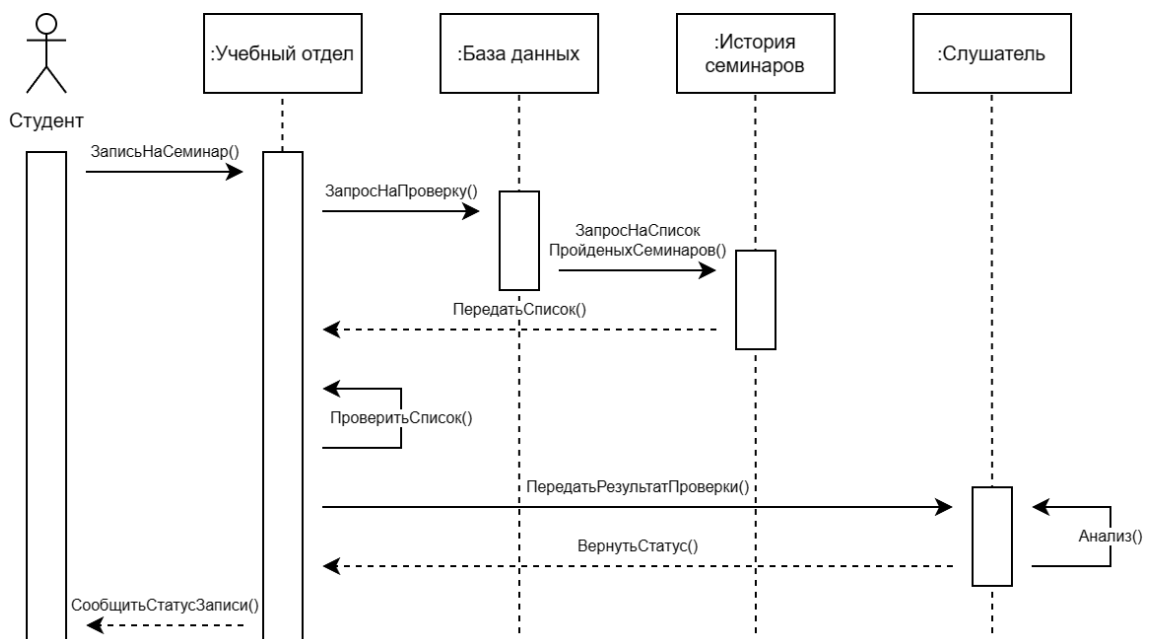


Рисунок 1 – Диаграмма последовательности

Таблица 1 – Взаимодействие элементов диаграммы

Отправитель	Тип сообщения	Наименование	Получатель
Студент	Синхронное	ЗаписьНаСеминар()	Учебный отдел
Учебный	Синхронное	ЗапросНаПроверку()	База данных
База данных	Синхронное	ЗапросНаСписокПройденныхСеминаров()	История семинаров
История семинаров	Возвращающее	ПередатьСписок()	Учебный отдел
Учебный отдел	Самовывоз	ПроверитьСписок()	Учебный отдел
Учебный отдел	Синхронное	ПередатьРезультатПроверки()	Слушатель
Слушатель	Возвращающее	ВернутьСтатус()	Учебный отдел
Учебный отдел	Возвращающее	СообщитьСтатусЗаписи()	Студент

Задание 2

Построить диаграмму кооперации по описанию приведенного варианта использования в п.1.

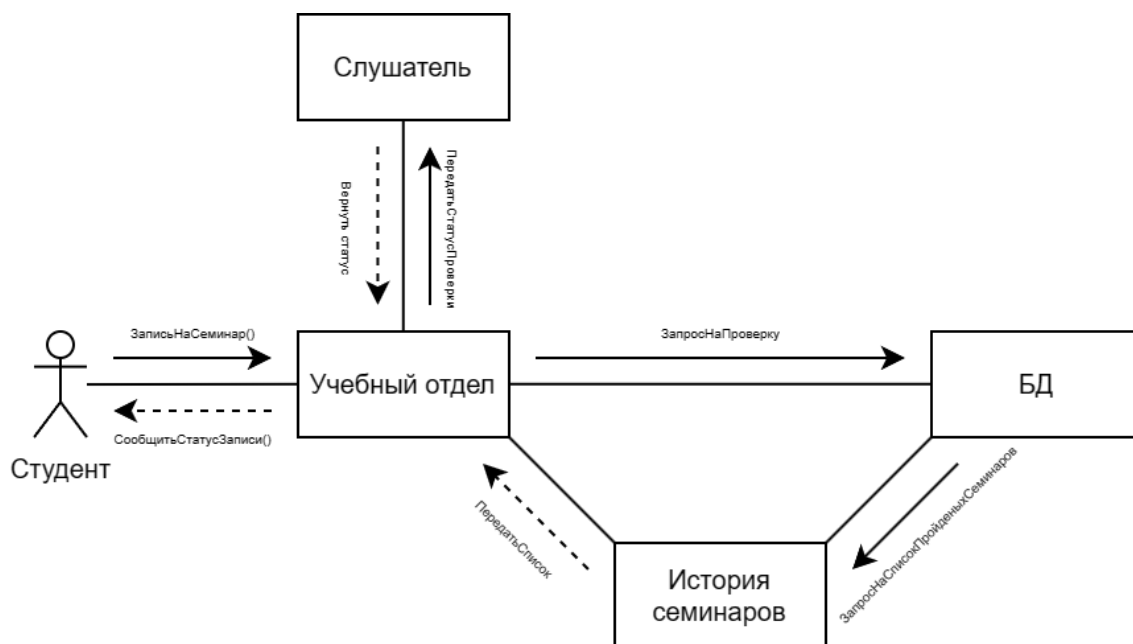


Рисунок 2 – Диаграмма кооперации

Задание 3

Построить модель отношений между объектами (диаграмма последовательности) рассматриваемой системы (варианта учебного проекта) в рамках одного прецедента.

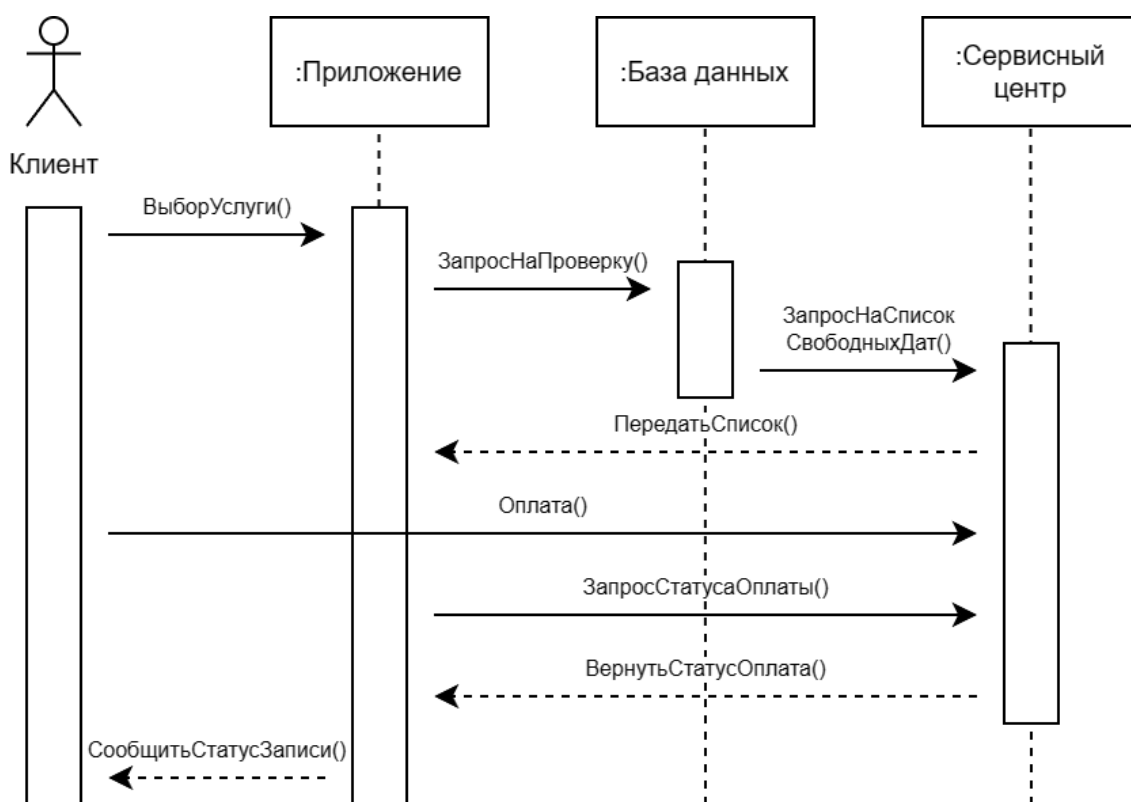


Рисунок 3 – Диаграмма последовательности для индивидуального задания

Задание 4

Построить модель отношений между объектами (диаграмма

кооперации) рассматриваемой системы (варианта учебного проекта) в рамках одного прецедента.

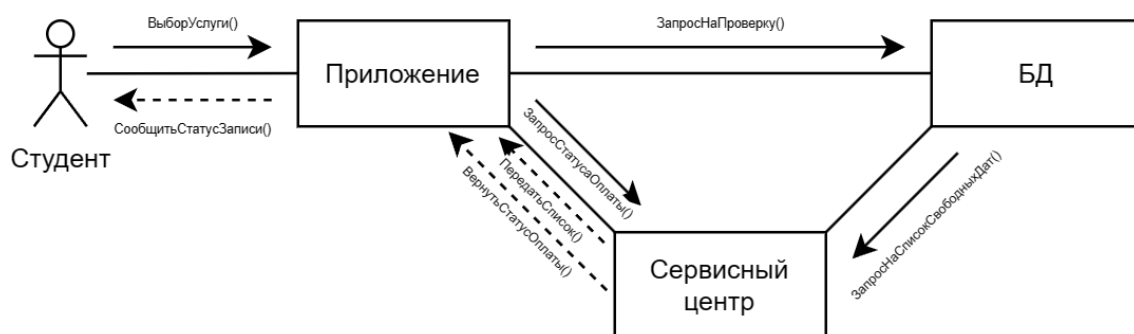


Рисунок 4 – Диаграмма кооперации для индивидуального задания

Отчет по практической работе № 5
Тема: «Построение UML – модели системы. Диаграмма классов.»

Цель работы

Изучить структуру модели проектирования, правила построения диаграммы классов.

Вариант

29 => Моделирование работы сервисного центра.

Теоретическое введение

Диаграмма классов представляет собой логическую модель статического представления моделируемой системы. Задача заключается в том, чтобы представить поведение более детально на логическом уровне.

Описание методологии моделирования классов в языке UML.

Объект представляет собой экземпляр класса – особую сущность, которая имеет заданные значения атрибутов и операций.

Атрибуты

Атрибут – это свойство класса. Атрибуты описывают перечень значений, в рамках которых указываются свойства объектов (т.е. экземпляров) этого класса. Класс может не иметь атрибутов или содержать любое их количество. Имена атрибутов, состоящие из одного слова, принято обозначать строчными буквами. Если имя состоит из нескольких слов, то эти слова объединяются, и каждое слово, за исключением первого, начинается с прописной буквы. UML позволяет отображать дополнительную информацию об атрибутах. В изображении класса можно указать тип для каждого значения атрибута. Перечень возможных типов включает строку, число с плавающей точкой, целое число, логическое значение и другие перечислимые типы. Для отображения типа используется двоеточие, которое отделяет имя атрибута от его типа. Здесь же можно указать значение атрибута по умолчанию.

Операции

Операция – это то, что может выполнять класс, либо то, что вы (или

другой класс) можете выполнять над данным классом. Подобно имени атрибута, имя операции записывается строчными буквами, если это одно слово. Если имя состоит из нескольких слов, они соединяются, и все слова, кроме первого, пишутся с прописной буквы. Список операций начинается ниже линии, отделяющей операции от атрибутов.

Помимо дополнительной информации об атрибутах, можно отобразить дополнительную информацию об операциях. В скобках, следующих за именем операции, можно указать параметр операции и его тип. Один из типов операций, *функция*, по окончании работы возвращает значение. В этом случае можно указать возвращаемое значение и его тип.

Для ассоциации, агрегации и композиции может указываться кратность (англ. multiplicity), характеризующая общее количество экземпляров сущностей, участвующих в отношении. Она, как правило, указывается с каждой стороны отношения около соответствующей сущности.

Кратность может указываться следующими способами:

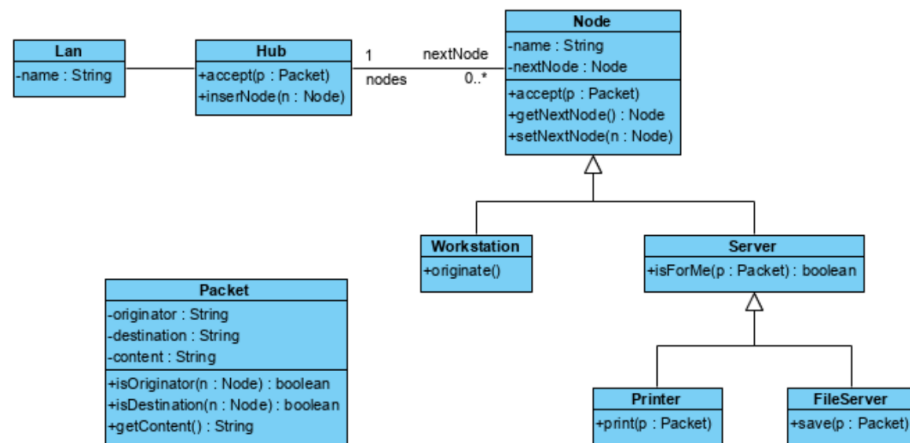
- * – любое количество экземпляров, в том числе и ни одного;

- целое неотрицательное число – кратность строго фиксирована и равна указанному числу (например: 1, 2 или 5);

- диапазон целых неотрицательных чисел "первое число .. второе число" (например: 1..5, 2..10 или 0..5);

- диапазон чисел от конкретного начального значения до произвольного конечного "первое число .. *" (например: 1..*, 5..* или 0..*);

- перечисление целых неотрицательных чисел и диапазонов через запятую (например: 1, 3..5, 10, 15..*).



При разработке диаграммы следует придерживаться следующих правил:

1. За основу диаграммы классов при ее разработке берется диаграмма классов анализа.

2. Для классов должны быть определены и специфицированы все атрибуты и методы. Их спецификация, как правило, выполняется с учетом выбранного языка программирования.

3. При определении методов рекомендуется использовать сообщения с ранее разработанных диаграмм последовательности и коммуникации.

4. Детальное проектирование граничных классов, как правило, не требуется. Большинство современных средств разработки поддерживает визуальную разработку интерфейса системы – меню, диалоговых форм, элементов диалоговых окон, панелей инструментов и т. д. В качестве исходных данных для их проектирования служат прототипы пользовательских интерфейсов. В связи с этим при проектировании таких классов основное внимание следует уделять особенностям отображения информации и специфичным операциям, которые возникают при диалоге пользователя с системой. Граничные классы, определяющие интерфейс взаимодействия с другими системами, требуют детального проектирования.

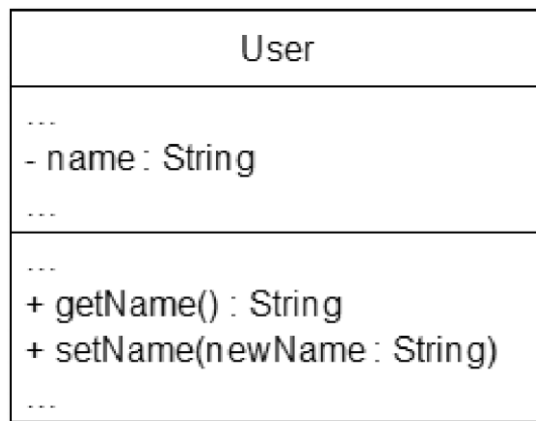
5. Для проектирования классов-сущностей можно применять подходы, используемые при проектировании БД, особенно в том случае, если данные будут храниться в таблицах БД. Если представление данных в БД и классах отличается друг от друга и в качестве хранилища информации будет применяться реляционная база данных, то рекомендуется разработать

отдельную диаграмму классов, описывающую состав и структуру БД.

6. Несмотря на то, что каждому объекту при выполнении программы автоматически назначается уникальный идентификатор, рекомендуется для классов-сущностей явно определять атрибуты, хранящие значения первичного ключа.

7. В отличие от реляционных БД поощряется использование в классах многозначных атрибутов в виде массивов, множеств, списков и т. д.

8. Управляющие классы следует проектировать только в случаях крайней необходимости – управления сложным взаимодействием объектов, реализации сложной бизнес-логики и вычислений, контроля целостности объектов и т. п. В противном случае функциональность этого класса лучше распределить между соответствующими граничными классами и классами сущностями.



Пример спецификации закрытого атрибута и методов для работы с ним

9. Ввиду большого количества классов в системе рекомендуется диаграммы классов разрабатывать отдельно для каждого пакета.

Решение

Задание 1

Построить диаграмму классов рассматриваемой системы (вариант учебного проекта)

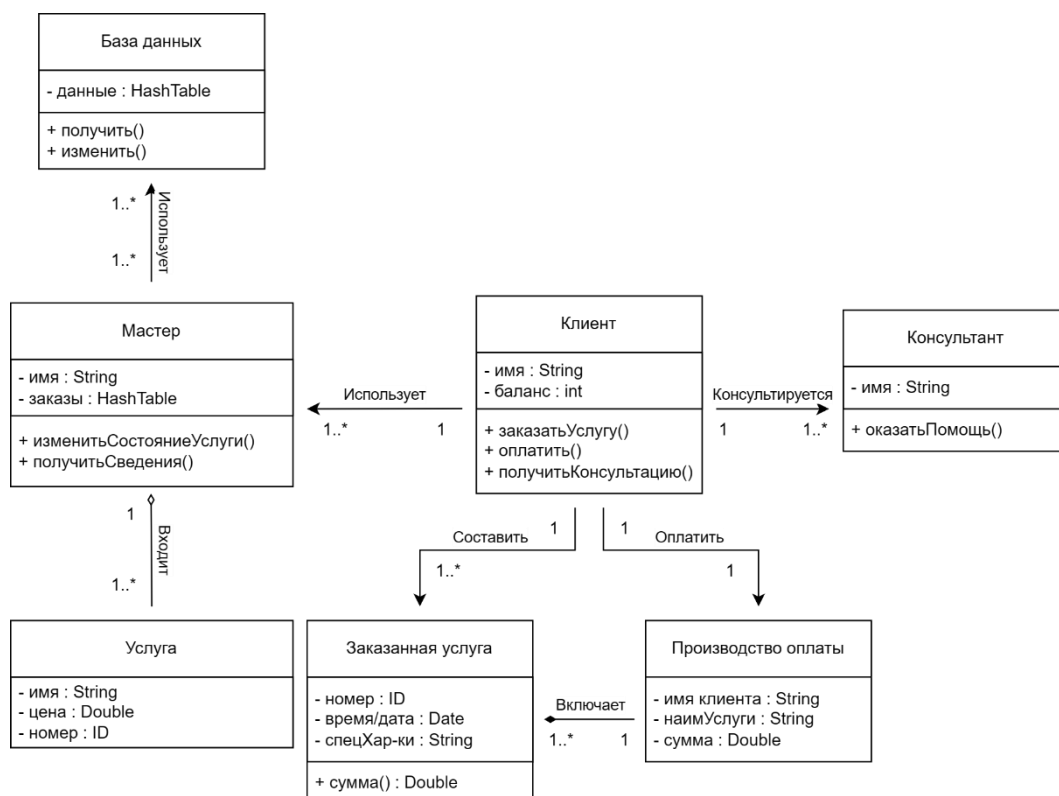


Рисунок 1 – Диаграмма классов работы сервисного центра

Задание 2

Заполнить таблицы

1,2 на основе полученной диаграммы в п.1

Таблица 1 — Описание классов диаграммы

Название класса	Описание
База данных	Класс, описывающий базу данных. Она включает в себя данные о списке покупателей, предоставляемых услугах и свободном времени мастеров. Эти данные можно получать и изменять.
Мастер	Класс, описывающий базу данных. Она включает в себя данные о имени мастера и данные о его заказах. Данный класс может изменять статус услуги и получать дополнительные сведения.
Услуга	Класс, описывающий услугу. Включает в себя название,

	цену и id услуги.
Клиент	Класс, описывающий клиента. Включает в себя данные о имени и баланс пользователя. Может запросить консультацию, заказать услуги и оплатить их.
Консультант	Класс, описывающий консультанта. Включает в себя данные о имени консультанта. Позволяет оказать помощь клиенту.
Заказная услуга	Класс описывающий список заказанных услуг. Включает в себя уникальный номер заказа, время и дату подачи заказа, а также специальную информацию, оставленную пользователем.
Производство оплаты	Класс, описывающий производство оплаты пользователем. Включает в себя информацию о имени клиента, названия оказываемой услуги, а также общую стоимость.

Таблица 2 — Взаимодействие между классами

Класс	Кратность	Тип отношения	Класс
Услуга	1..* : 1	Агрегация	Мастер
Мастер	1..* : 1..*	Ассоциация	База данных
Клиент	1 : 1..*	Ассоциация	Мастер
Клиент	1 : 1..*	Ассоциация	Заказанная услуга
Клиент	1 : 1	Ассоциация	Производство оплаты
Клиент	1 : 1..*	Ассоциация	Консультант
Производство оплаты	1 : 1..*	Композиция	Заказанная услуга

Отчет по практической работе № 6
Тема: «Построение UML – модели системы. Диаграмма деятельности.»

Цель работы

Научиться строить усовершенствованные блок-схемы с параллельными процессами.

Вариант

29 => Моделирование работы сервисного центра.

Теоретическое введение

При моделировании поведения системы возникает необходимость детализировать особенности алгоритмической и логической реализации выполняемых системой операций.

Для моделирования процесса выполнения операций в языке UML используются так называемые диаграммы деятельности. Каждое состояние на диаграмме деятельности соответствует выполнению некоторой элементарной операции, а переход в следующее состояние срабатывает только при завершении этой операции в предыдущем состоянии.

Графически диаграмма деятельности представляется в форме графа деятельности, вершинами которого являются состояния действия, а дугами - переходы от одного состояния действия к другому. На диаграмме деятельности отображается логика или последовательность перехода от одной деятельности к другой, при этом внимание фиксируется на результате деятельности. Компонентами диаграммы деятельности являются:

- состояния действия,
- переходы,
- дорожки,
- объекты.

Состояние действия (action state) является специальным случаем состояния с некоторым входным действием и, по крайней мере, одним выходящим из состояния переходом. Этот переход неявно предполагает, что

входное действие уже завершилось. Состояние действия не может иметь внутренних переходов, поскольку оно является элементарным.

Переходы. При построении диаграммы деятельности используются только нетриггерные переходы, т. е. такие, которые срабатывают сразу после завершения деятельности или выполнения соответствующего действия. На диаграмме такой переход изображается сплошной линией со стрелкой.

Дорожки. Диаграммы деятельности могут быть использованы для моделирования бизнес-процессов. Применительно к бизнес-процессам желательно выполнение каждого действия ассоциировать с конкретным подразделением компании. В этом случае подразделение несет ответственность за реализацию отдельных действий, а сам бизнес-процесс представляется в виде переходов действий из одного подразделения к другому.

Для моделирования этих особенностей в языке UML используется специальная конструкция, получившая название дорожки (swimlanes). При этом все состояния действия на диаграмме деятельности делятся на отдельные группы, которые отделяются друг от друга вертикальными линиями. Две соседние линии и образуют дорожку, а группа состояний между этими линиями выполняется отдельным подразделением.

В общем случае действия на диаграмме деятельности выполняются над теми или иными объектами. Эти объекты либо инициируют выполнение действий, либо определяют некоторый результат этих действий. При разработке диаграммы следует придерживаться следующих правил:

Количество пересечений линий следует минимизировать. При этом считается, что пересекающиеся линии не имеют логической связи друг с другом. Другими словами, потоки данных или управления в местах пересечений не меняют своего направления.

Если на диаграмме имеется ветвление / решение на параллельные или альтернативные потоки, то должно указываться и соответствующее соединение / слияние этих потоков.

При использовании альтернативных потоков каждый из них должен быть

специфицирован с помощью сторожевого условия. Сторожевые условия не должны допускать одновременного срабатывания двух и более переходов.

Решение

Задание 1

Описать возможные последовательности состояний и переходов, которые характеризуют поведение элемента исследуемой системы с помощью диаграммы состояний (индивидуальный вариант учебного проекта).

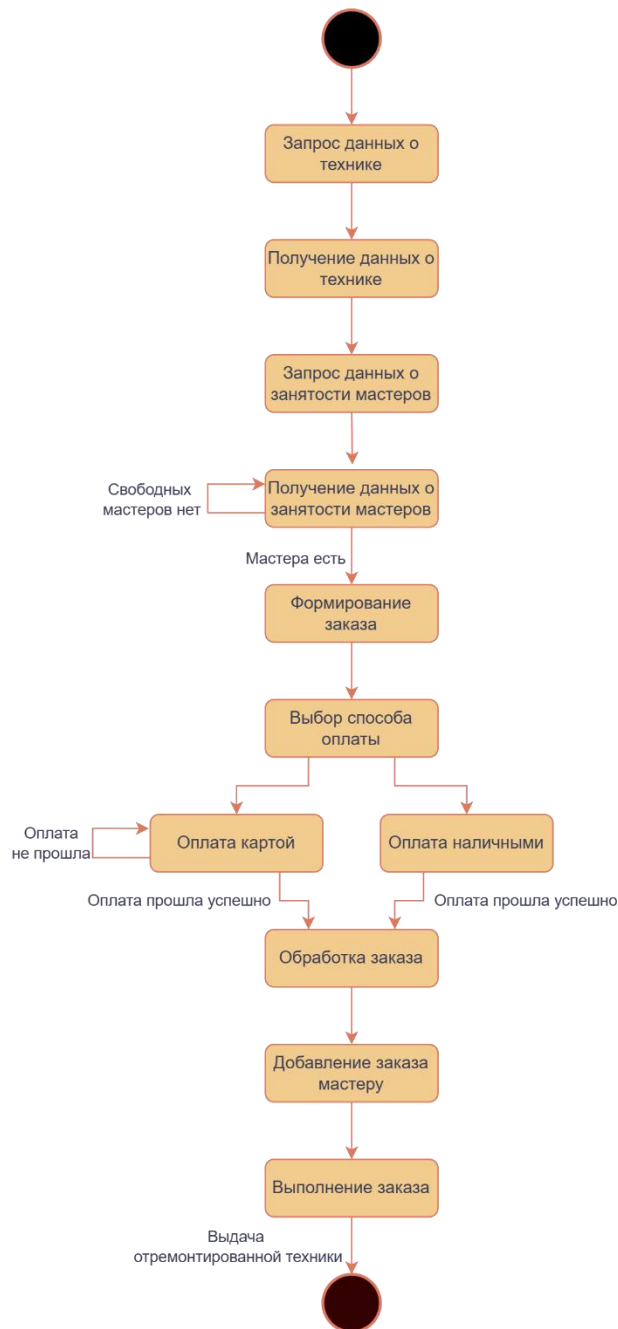


Рисунок 1 – Диаграмма состояний для моделирования организаций

Сначала должен быть сформирован запрос данных о технике, далее рассматривается проверяются свободные мастера, после чего формируется заказ. После формирования его необходимо оплатить, если оплата прошла, то заказ отправляется на обработку, затем мастеру, а после уже отдается заказчику.

Задание 2

Описать все системные операции посредством диаграммы деятельности.

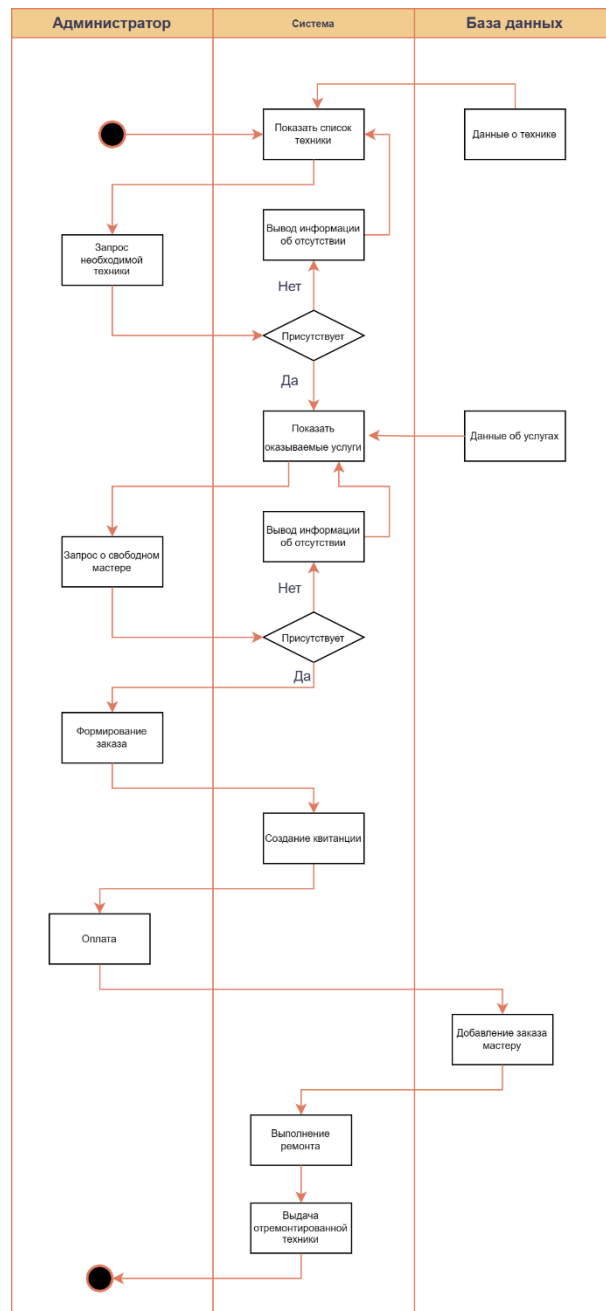


Рисунок 2 – Диаграмма деятельности

Сначала должен быть сформирован запрос данных о технике, далее рассматриваются данные об необходимой технике, а также о свободных мастерах. После формирования квитанции ее необходимо отправить и оплатить, если оплата прошла, то заказ отправляется на обработку, затем заказ отправляется мастеру, а после уже отдается заказчику.

Отчет по практической работе № 7
Тема: «Построение UML – модели системы. Диаграммы компонентов, Развертывания.»

Цель работы: научиться строить модель реализации.

Задачи: построить модель реализации с помощью диаграмм компонентов и развертывания с рассмотрением основных элементов и правил построения.

ПО: Visual Paradigm, Draw.io, Rational Rose.

Порядок выполнения работы:

1. Построить диаграмму компонентов организации сервиса по починке компьютеров.
2. Построить диаграмму развертывания рассматриваемой системы.

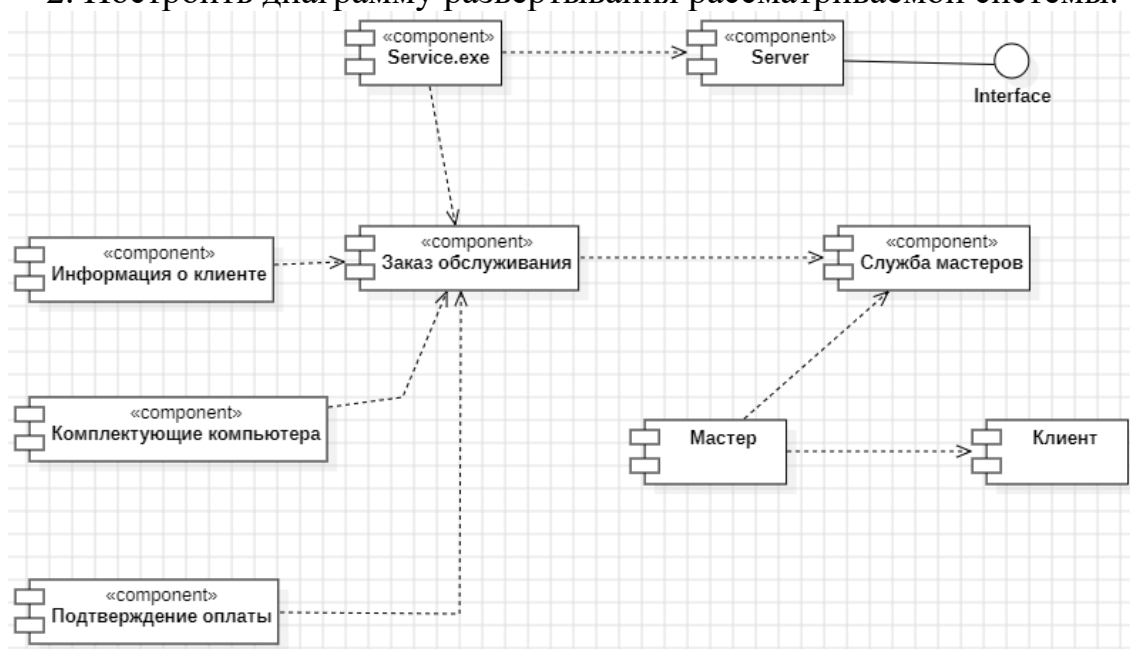


Рисунок 1 – Диаграмма компонентов организации сервиса по починке компьютеров

На данной диаграмме компонентов представлено взаимодействие сервиса ремонта компьютеров с клиентом. Чтобы осуществить заказ ремонта необходимо сформировать заказ об обслуживании в приложение, которое в дальнейшем передает информацию о заказе на сервер. После выполнения заказа мастером отремонтированный компьютер отдается клиенту.

2. Построить диаграмму развертывания рассматриваемой системы.

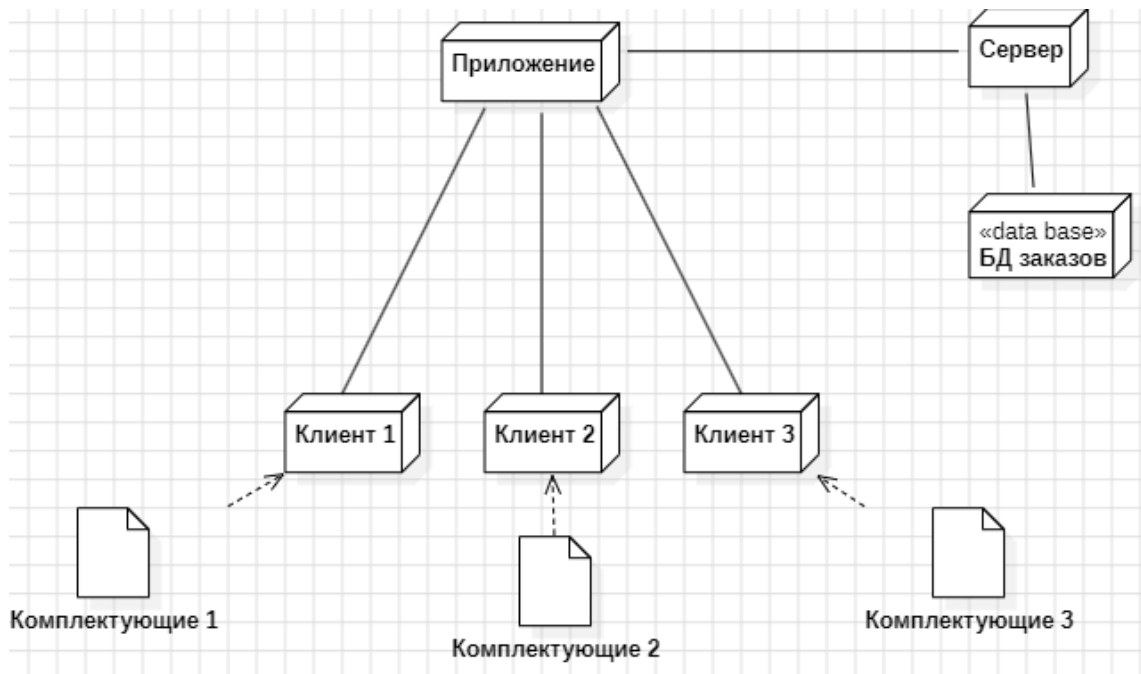


Рисунок 2 - Диаграмма развертывания

Клиент делает заказ через систему формирования заказа с помощью приложения. После формируется заказ, который в последующем отправляется на сервер. Сервер обрабатывает запрос и добавляет информацию в БД.

Доклад

Тема: «Современные подходы имитационного моделирования.»

Имитационное моделирование позволяет нам создавать компьютерные модели для изучения сложных систем и процессов, что важно для принятия обоснованных решений.

Применение

Имитационное моделирование активно используется во многих областях, включая экономику, транспорт и здравоохранение. Оно помогает анализировать поведение системы, оптимизировать процессы и принимать эффективные решения.

Современные подходы включают в себя:

- **Агентно-ориентированное моделирование (АОМ)** фокусируется на взаимодействии агентов в системе.
- **Мультиагентные системы (МАС)** расширяют АОМ, позволяя множеству агентов взаимодействовать друг с другом.
- **Имитационное моделирование на основе искусственного интеллекта (ИИ)** использует ИИ для создания более точных моделей.
- **Гибридные подходы** сочетают различные методы моделирования для улучшения результатов.

АОМ исследует поведение децентрализованных агентов и их влияние на систему в целом. Это подход "снизу вверх", где глобальное поведение системы возникает из взаимодействия индивидуальных агентов.

МАС — это системы, где независимые агенты сотрудничают для достижения общих целей. Каждый агент имеет свои цели и информацию, на основе которой он принимает решения. МАС применяются в различных сферах, от онлайн-торговли до управления чрезвычайными ситуациями.

Имитационное моделирование на основе ИИ объединяет ИИ и моделирование для создания виртуальных моделей, учитывающих

взаимодействие агентов и динамику системы. Это позволяет создавать более точные и адаптивные модели.

Гибридные подходы комбинируют разные методы, например, АОМ с системной динамикой или ИИ для оптимизации параметров модели, что улучшает качество прогнозов и анализа систем.

Заключение

При выполнении всех практических работ были изучены: структура и функционал рассматриваемой системы, элементы и правила построения диаграммы вариантов использования, структура иерархии классов системы, структура анализа и правила построения диаграмм последовательности и кооперации, модель проектирования и правила построения диаграмм классов. Также были получены знания о построении усовершенствованных блок-схемы с параллельными процессами и о построении модели реализации.