**Программа учебной дисциплины**

**«Проектирование архитектуры программных систем»**

Утверждена

Академическим советом ООП

Протокол № от «\_\_»\_\_\_\_\_20\_\_ г.

|  |  |
| --- | --- |
| Автор  | Кумсков М. И., проф., д. физ.-мат. н. |
| Число кредитов  | 5 |
| Контактная работа (час.)  | 64 |
| Самостоятельная работа (час.)  | 126 |
| Курс  | 3 |
| Формат изучения дисциплины | Без использования онлайн-курсов |

1. **ЦЕЛЬ, РЕЗУЛЬТАТЫ ОСВОЕНИЯ ДИСЦИПЛИНЫ И ПРЕРЕКВИЗИТЫ**

Целями освоения дисциплины «Проектирование архитектуры программных систем» являются получение теоретических знаний и практических навыков работы с требованиями к информационным системам (ИС) как входных данных для проектирования, а также получение теоретических знаний и практических навыков при выполнении основных задач архитектора и дизайнера в процессе проектирования и визуального моделирования на UML, согласно методологии IBM Rational Unified Process (IBM RUP).

В результате освоения дисциплины студент должен:

**знать:**

* основные понятия, используемые при проектировании и создании приложений, информационных систем (ИС), включая лучшие практики разработки программного обеспечения(ПО);
* основы визуального моделирования при проектировании и разработке архитектуры ПО, способы преобразования требований к ИС на основе сценариев использования в архитектурные представления ПО;
* основные задачи, выполняемые архитектором и разработчиком при проектировании по согласно методологии IBM RUP.

**уметь:**

* строить проектные модели ПО и ИС с использование диаграмм визуальной нотации UML (Унифицированный язык моделирования);
* проектировать логическое представление архитектуры на UML;
* пошагово строить на UML диаграммы классов, представляющие модель предметной области и ключевые абстракции проекта;
* с использование паттернов проектирования трансформировать модель предметной области;
* идентифицировать и разрабатывать спецификации сценариев использования ИС для создания классов-участников реализации;
* использовать диаграммы взаимодействия, состояний и активности при проектировании ПО.

**владеть:**

* навыками пошагового построения проекта ПО как последовательного набора UML диаграмм в инструментальном CASE средстве;
* навыками реализации лучших практик разработки программного обеспечения, включая построения компонентной архитектуры ПО и применение архитектурных механизмов анализа;
* навыками использования визуальных диаграмм UML при проектировании архитектурных представлений.

Изучение дисциплины базируется на следующих дисциплинах:

* «Программирование»
* «Введение в программную инженерию»
* «Конструирование программного обеспечения»
* «Алгоритмы и структуры данных»

Для освоения учебной дисциплины студенты должны владеть следующими знаниями и компетенциями:

* Владеть: навыками решения задач из следующих разделов современной математики: математический анализ, линейная алгебра и аналитическая геометрия, математическая логика, дискретная математики, знаниями основных алгоритмов при работе с массивами, списками и деревьями, навыками разработки программ и программных систем с использованием баз данных.
* Знать: основные направления, проблемы, теории и методы современной математики, в том числе основы математического анализа, линейной алгебры, дискретной математики и математической логики, основные понятия информатики, используемые при проектировании и разработке программ.
* Уметь: решать стандартные задачи математического анализа, линейной алгебры, дискретной математики и математической логики и применять идеи, использованные в их решениях, для решения аналогичных задач, составлять решения задач на алгоритмических языках, включая Си и С++.
* Иметь навык: программирования на языке высокого уровня типа Си++ или С#.

Основные положения дисциплины должны быть использованы в дальнейшем при подготовке курсовых и выпускных квалификационных работ.

1. **СОДЕРЖАНИЕ УЧЕБНОЙ ДИСЦИПЛИНЫ**

**Тема 1. Методология IBM Rational Unified Process (IBM RUP). Лучшие практики разработки ИС.**

Назначение лучших практик. Связь практик с визуальным моделированием. Инструментальная поддержка практик. Понятие компоненты в архитекторе ИС. Интерфейс компоненты. Версионный контроль основных документов. Понятие конфигурации. Понятие архитектуры ИС, архитектурные представления ИС.

**Тема 2. Унифицированный язык моделирование UML. Диаграммы UML.**

Основные принципы визуального моделирования. Сложность ПО и архитектурные представления.

Статические и динамические диаграммы UML. Репозиторий модели CASE средства. Прямое и об-

ратное проектирование кода программ и структуры базы данных. Стереотипы UML и их использование.

**Тема 3. Классы и объекты. Отношения между классами. Диаграмма классов UML.**

Понятие класса и объекта. Отношения между классами и из визуальное представление. Ассоциация и ее разновидности. Понятие навигации ассоциации, наследования, зависимости. Отличие агрегации от композиции. Понятие множественности ассоциации. Паттерны при создании и преобразовании UML диаграмм классов. Ассоциативные классы и их использование в проекте.

**Тема 4. Процесс выявления требований к ИС (IBM RUP). Сценарии использова-**

**ния ИС (СиС).**

Заинтересованные лица, свойства ИС, классификация свойств. Формулировка проблемы. Диаграмма причинно-следственных связей. Матрица трассировки требований. Техника выявления трудоемкости СиС и свойств ИС. Понятие риска. Понятие объема работ проекта (Scope). Инструментальная поддержка процесса.

**Тема 5. Эскиз и спецификация сценария использования. Диаграмма СиС UML.**

Известные подходы к их идентификации. Первичные и вторичные экторы. Эскиз и спецификация Сценария Использования. Выявление СиС в бизнес-процессах. Структура спецификации. Основной и альтернативные потоки. Паттерны выявления Сценариев использования ИС. Учет бизнес правил в СиС. Пред-условия и пост-условия СиС. Визуальное представление на UML модели сценариев использования. Словарь проекта, Концепция ИС, модель Сценариев Использования, спецификация.

**Тема 6. Модель предметной области. Пошаговое построение. Паттерны проектирования.**

Процесс формирования визуальной модели. Регистрируемые события и объекты. Связь модели предметной области с задачей идентификации функциональных требований к ИС. Используемые UML диаграмм, паттерны при их построении. Понятие состояния объекта и его визуальное представление. Поиск экторов на модели предметной области.

**Тема 7. Анализ и проектирование – дисциплина IBM Rational Unified Process.**

Обзор дисциплины – входные и выходные артефакты (рабочие материалы) дисциплины. Роли и задачи дисциплины. Назначение и задачи этапов дисциплины. Используемые диаграммы UML в задачах проектирования.

**Тема 8. Этап анализа. Задача архитектора: «Архитектурный анализ».**

Роль архитектора в проекте и выполняемые им задачи. Ключевые абстракции и их идентификация. Диаграммы UML для визуализации ключевых абстракций. Архитектурные механизмы, их назначение. Архитектурные паттерны и их использование. Понятие слоя и его представление на UML. Структура «реализация сценария использования» (use case realization) и ее представление в визуальной модели.

**Тема 9. Архитектурные представления. Пакеты на UML. Ключевые абстракции.**

Понятие заинтересованных лиц и детальности проекта. Архитектурные паттерны и их использование в дизайне. Использование модели предметной области в задаче. Связи между пакетами UML и их использование в представлении архитектуры.

**Тема 10. Этап анализа. Задача разработчика: «Анализ сценария использования».**

Выявление классов участников. Применение сценариев использования для проектирования объектной динамики. Стереотипы «boundary» «control» «entity» классов-участников реализации. Паттерны для идентификации классов.

**Тема 11. Объектная динамика при проектировании. Диаграммы взаимодействия UML.**

Диаграммы последовательности и коммуникация. Правила использования стереотипов «boundary» «control» «entity» при проектировании объектной динамики.

**Тема 12. Диаграмма «Представление классов участников». Ответственности классов.**

Правила идентификации операций (ответственности) классов. Правила идентификации отношений между классами – ассоциаций и зависимостей.

**Тема 13. CASE средства визуального моделирования. Прямое и обратное проектирование.**

Различия рисования и визуального моделирования. Репозиторий CASE средства и синхронизация его содержимого на UML диаграммах. Понятие каркасного кода при прямом проектировании. Сравнение возможностей CASE средства визуального моделирования

**Тема 14. Этап дизайна. Задача архитектора «Идентификация проектных элементов».**

Определение классов, подлежащих декомпозиции. Упаковка концептуальных классов по пакетам проекта. Зависимости между пакетами. Архитектурный паттерн «Слои» и его использование в задаче.

**Тема 15. Этап дизайна. Задача архитектора «Идентификация проектных механизмов».**

Понятие паттерна. Представление паттернов в CASE средстве. Типы архитектурных механизмов и их представление в визуальное модели.

**Тема 16. Этап дизайна. Задача разработчика «Проектирование сценария использования».**

Применение архитектурных механизмов. Представление компонент в дизайне. Использование интерфейсов на диаграммах последовательности UML.

**Тема 17. Этап дизайна. Задача разработчика «Проектирование подсистем».**

Структурный класс UML и его использование. Операции интерфейса и их реализация в подсистеме. Диаграмма последовательности UML. Динамическое и статическое представление дизайна подсистемы. Использование механизмов проектирования. Зависимости классов и пакетов.

**Тема 18. Этап дизайна. Задача разработчика «Проектирование классов». Диаграмма состояний на UML.**

Ответственности и операции классов. Выбор атрибутов классов. Атрибуты и понятие состояния объекта. Паттерны выявления состояний. Специальные состояния – начальное и конечное. Суперсостояние и его использование. Условия на диаграмме состояний. Правила перехода из состояния в состояния и из запись на диаграмме.

1. **ОЦЕНИВАНИЕ**

Преподаватель оценивает работу студентов. Накопленная оценка учитывает результаты студента по текущему контролю следующим образом: Онакопленная = 0.5\*Опроект + 0.5\*Одз,

где Одз – это среднее значение оценки по всем домашним заданиям (ДЗ): Одз принимает

значение от 0 до 10. Если какое-то ДЗ не было выполнено, то за него ставится оценка 0.

Опроект – оценка за проектную работу, выполненную по заданной постановке задачи. В ходе проекта следует выполнить следующие этапы, каждый из которых оценивается отдельно в интервале от 0 до 10:

1. выявить требования к информационной системе (ИС) в виде сценариев использования системы (СиС) и модели предметной области – построить соответствующие диаграммы на UML – оценка О(1);
2. подготовить спецификации СиС по известному шаблону, включая основной и альтернативные потоки событий, предусловия и пост-условия – оценка О(2);
3. идентифицировать классы участники реализации СиС - граничные классы, управляющие и классы-сущности, - и распределить по ним поведение СиСа для каждого потока событий, построив соответствующие диаграммы взаимодействия на UML – оценка О(3);
4. построить диаграмму классов на UML для классов участников реализации СиСа – оценка О(4).

Проектирование ИС на UML проводится с использованием CASE средства, выбранного для проведение работ.

В результате Опроект = (О(1) + О(2) + О(3) + О(4)) / 4.

Величина оценки Онакопленная может варьироваться от 0 до 10.

Результирующая оценка за дисциплину рассчитывается следующим образом:

Орезультирующая = 0.4\*Онакопленная + 0.6 \*Оэкзамен

Способ округления результирующей, накопленной, экзаменационной оценок по учебной дисциплине – арифметический. Оценки по всем формам текущего контроля выставляются по 10-ти балльной шкале.

1. **ПРИМЕРЫ ОЦЕНОЧНЫХ СРЕДСТВ**

# **Оценочные средства для текущего контроля студента**

Далее приводятся примерные задания для проектной работы / экзамена, домашних работ.

Пусть задана постановка задачи (см. примеры постановок ниже).

Необходимо:

1. выявить требования к информационной системе (ИС) в виде сценариев использования системы (СиС) и построить соответствующие диаграммы на UML,
2. найти классы модели предметной области –– и построить соответствующие диаграммы на UML,
3. подготовить спецификацию заданного СиС по известному шаблону, включая основной и альтернативные потоки событий, пред-условия и пост-условия,
4. идентифицировать классы участники реализации заданного СиС- граничные классы, управляющий и классы сущности, - и распределить по ним поведение СиСа для каждого потока событий, построив соответствующие диаграммы взаимодействия на UML,
5. построить диаграмму классов на UML для классов участников реализации СиСа.

Примеры постановок задачи (в ходе реализации дисциплины список постановок задач пополняется):

Задача 1. «Автоматизация поликлиники»

Районный отдел здравоохранения принял решение разработать ИС для учета заболеваемости в районе, загруженности медицинского персонала в поликлиниках и в районной больнице. Информационная система должна поддерживать ответы на следующие запросы (за текущий период):

• Сколько пациентов обратилось в поликлиники района?

• По каждому типу болезни, сколько новых диагнозов поставлено?

• Сколько больничных листов выдано по данному типу заболевания?

• Сколько койко-мест свободно (занято) в каждом отделении районной больнице на данное число?

• Какова загруженность врачей данной специальности по поликлиникам района?

В поликлиниках ведется запись пациентов к врачам-специалистам, ИС должна поддерживать эту работу регистратуры – учитывать приемы у врача. На приеме у врача пациенту ставится диагноз по его жалобе. На приеме пациенту может быть выписан (открыт) или закрыт больничный лист, также может быть выписано направление на лечение в районную (областную) больницу. По результату приема пациента медицинская сестра в регистратуре (по документам, предоставленным врачом, в конце рабочего дня) или сам врач (в своем кабинете) вносит в ИС данные о приеме данного пациента.

По направлению на лечение проводится прием в больницу, пациент размещается в соответствующем (диагнозу) отделении больницы, ему выделяется больничная койка (койко-место). После прохождения лечения пациент выписывается из больницы и направляется в поликлинику, где ему закрывают (или продлевают) больничный лист. ИС должна регистрировать поступление и выписку пациентов по отделениям больницы.

Задача 2. «Комбинат питания»

Предприятие владеет сетью небольших кафе и ресторанов – пунктов питания (ПП), - по всему городу. Руководство решило ввести централизованный учет движения денежных средств, включая возможность расчета средней стоимости чека как характеристики пункта питания.

Основной услугой ПП является простое обслуживание клиента. Клиент заказывает, используя меню, состав своего обеда (ужина) как набора блюд и напитков, официант проводит обслуживание, выписывает чек, клиент оплачивает чек (наличными) и по своему усмотрению может оставить чаевые официанту. Каждый ПП самостоятельно проводит закупку продуктов для своих нужд. Согласно рецепту на приготовление каждого блюда расходуется определенный состав продуктов (мясо, рыба, овощи, фрукты, специи, масла...). За состав блюд и рецептуру отвечает шеф-повар данного кафе. Себестоимость блюда складывается из состава продуктов по рецепту и времени повара и его помощников при приготовлении блюда. Цена блюда определяется как его себестоимость умноженная на ценовой коэффициент, фиксированный для данного ресторана (кафе).

Рестораны и кафе могут проводить (по предварительному заказу клиента) «корпоратив»: празднование юбилея, организацию свадебного ужина, праздника, корпоративного мероприятия. В этом случае кафе закрыто для обычных посетителей и арендная плата включается в себестоимость мероприятия. Для каждого корпоратива составляется собственный список блюд, под который специально закупаются продукты и напитки. Кроме стоимости блюд и аренды помещения стоимость корпоратива определяется составом официантов, обслуживание гостей, и составом поваров, проводящих приготовление блюд. Корпоратив при заказе оплачивается авансовым платежом в кассу ПП, а по завершения мероприятия лиенту (оформлявшего корпоратив) выставляется счет за полный состав услуг и блюд, который клиент оплачивает (за вычетом суммы аванса).

Внедряемая информационная система (ИС), имеющая «клиентские терминалы» в каждом кафе (ресторане), должна вести учет продуктов (для каждого ПП), потребляемых клиентами блюд, числа обслуженных клиентов, числа проводимых корпоративов и т.п. ИС должна иметь возможность формирование различных отчетов, включая расчет среднего чека по ПП, средней стоимости корпоратива, занятости персонала в ПП и на корпоративах, а также доходов и расходов как по ПП, так и по периодам времени.

Задача 3. «Документы муниципалитета»

Руководству муниципалитета требуется автоматизировать работу своей канцелярии в плане контроля исполнения решений, принятых по письмам граждан. Новая информационная система «ДОКУМЕНТ» должна предоставлять информацию о полученных письмах и запросах граждан, о принятых по ним решениях, об ответах и их датах, о постановке решения «на контроль». На любой запрос (письмо, жалоба, обращение, заявление и т.д.) сотрудники муниципалитета обязаны ответить в течении 10 календарных дней.

Система «ДОКУМЕНТ» должна вести регистрацию входящих писем, их классификацию (по внутренним правилам), назначение отдела, которому поручено заниматься письмом, назначение контрольного срока исполнения ответа. Класс письма присваивает письму канцелярия при его регистрации как входящего документа. Каждый отдел отвечает за обработку писем заданных ему классов.

После регистрации в системе «ДОКУМЕНТ» бумажные документы – с соответствующей пометкой: номер и дата, - передаются в отделы для обработки. В отделе его руководитель назначает исполнителя ответа, - одного из своих сотрудников, - и заносит это решение в систему «ДОКУМЕНТ». Начальник может просмотреть в системе «ДОКУМЕНТ» исполнение поручений по датам, сотрудникам, видам писем и распечатать соответствующий отчет.

По документу назначенный сотрудник готовит ответ, текст которого сохраняется в системе. После занесения ответа в систему канцелярия распечатывает ответ на бланке муниципалитета и подписывает документ в соответствующем отделе. После подписи ответ регистрируется как выходящий с номером и датой и передается в экспедицию муниципалитета для доставки на почтовое отделение. Сотрудник делает пометку в системе «ДОКУМЕНТ» о выполненном поручении по данному письму, и канцелярия подтверждается «закрытие письма». Исходные (поступившие) письма в систему не заносятся – ведется только учетная информация о их движении по отделам. Кроме простого письма-ответа возможны следующие действия по документам.:

• Подготовлен проект решения «необходимо проведение мероприятия муниципалитетом»,

• Подготовлен проект решения «необходимо проведение мероприятия смежными организациями».

Проект решения выносится на очередное заседание муниципалитета начальником соответствующего отдела. Если решение принято, то планируемые мероприятия по письмам должны быть поставлены канцелярией на контроль с указанием ответственного лица, на контроль ставится и само письмо в новой системе. Документ – обращение гражданина, - снимается с контроля при успешном завершении соответствующего мероприятия. Об этом событии готовится ответ заявителю, чье письмо послужило причиной назначения мероприятия. Ответ готовит тот же, сотрудник, который подготавливал первый ответ по письму и проект решения по проведению мероприятия.

Список сотрудников и список отделов новая система «ДОКУМЕНТ» должна брать из уже существующей кадровой системы муниципалитета «Кадры». Информация о планируемых мероприятиях (по которым приняты решения муниципалитета) передается в существующую систему «Поручения и проекты» муниципалитета, где ее средствами отслеживаются этапы мероприятия/проекта. После завершения мероприятия по системе «Поручения и проекты» канцелярия отслеживает это событие и снимает с контроля письмо в системе «ДОКУМЕНТ». Руководство муниципалитета имеет доступ к системе для получения отчетов и проведения оперативного поиска. Доступ пользователей к системе авторизован.

**Оценочные средства для промежуточной аттестации**

Примерный перечень вопросов к итоговому контролю (экзамену) для самопроверки студентов:

(1) Сервер приложений в архитектуре ИС. Построение информационной системы без сервера приложений.

(2) Многозвенная архитектура ИС. Ее представление в проекте.

(3) Итерационная и каскадная разработка ИС. Правила организации итераций. Назначение фаз при разработке ИС.

(4) Управление требованиями, основные создаваемые документы и их содержание.

(5) Лучшие практики разработки ИС.

(6) Опишите назначение лучших практик разработки программного обеспечения.

(7) Какая существует связь лучших практик разработки ПО с визуальным моделированием.

(8) Какие программные инструменты осуществляют поддержку лучших практик разработки ПО.

(9) Что такое компонента, как компоненты используются при проектировании ПО

(10) Опишите, что такое конфигурация сборки ПО.

(11) Определите понятие архитектуры ИС, что такое архитектурные представления ПО.

(12) Формирование требований к информационной системе (ИС).

(13) Что такое Функциональные требования к ПО? Приведите примеры.

(14) Что такое атрибуты качества ПО? Приведите примеры.

(15) Какие документы создаются при определении требований к ПО, каково их содержание.?

(16) Модель предметной области.

(17) Опишите шаги процесса формирования визуальной модели ПО.

(18) Кто использует регистрируемые события и объекты?

(19) Какова связь модели предметной области с задачей идентификации функциональных требований?

(20) Какие UML диаграммы создаются при создании модели предметной области, перечислите паттерны проектирования используются?

(21) Понятие состояния объекта и его визуальное представление на UML.

(22) Как можно определить экторов ИС по модели предметной области?

(23) Каковы подходы к идентификации первичных и вторичных экторов?

(24) Эскиз и спецификация Сценария Использования. Их состав, область использования, в чем сходство и различия?

(25) Выявление сценариев использования в бизнес-процессах.

(26) Структура спецификации Сценария Использования.

(27) Основной и альтернативные потоки Сценария Использования.

(28) Паттерны выявления Сценариев использования ИС.

(29) Учет бизнес правил в Сценариях Использования

(30) Пред-условия и пост-условия СиИС.

(31) Визуальное представление на UML модели сценариев использования.

(32) Процесс выявления требований к ИС.

(33) Заинтересованные лица, свойства ИС, классификация свойств.

(34) Формулировка проблемы, решаемой ИС в концепции ИС.

(35) Матрица трассировки требований и ее использование при работе с требованиями.

(36) Обзор дисциплины «Анализ и проектирование» – входные и выходные артефакты (рабочие материалы).

(37) Роли и задачи дисциплины «Анализ и проектирование». Назначение и задачи этапов дисциплины.

(38) Используемые диаграммы UML в задачах дисциплины «Анализ и проектирование».

(39) Задача «Архитектурный анализ». Роль и выполняемые шаги.

(40) Ключевые абстракции и их идентификация.

(41) Диаграммы UML для визуализации ключевых абстракций.

(42) Архитектурные механизмы, их назначение.

(43) Архитектурные паттерны и их использование.

(44) Понятие слоя и его представление на UML.

(45) Структура «реализация сценария использования» (use case realization) и ее представление в визуальной модели.

(46) Понятие класса и объекта. Диаграмма классов на UML.

(47) Отношения между классами и из визуальное представление.

(48) Понятие навигации, наследования, зависимости.

(49) Отличие агрегации от композиции.

(50) Понятие множественности ассоциации.

(51) Паттерны при создании и преобразовании UML диаграмм классов.

(52) Задача «Анализ сценария использования». Роль и выполняемые шаги.

(53) Концептуальные классы, их стереотипы и назначение в визуальной модели.

(54) Диаграммы взаимодействия UML и их назначение.

(55) Диаграмма последовательности и ее использование в задаче.

(56) Понятие ответственности класса.

(57) Диаграмма классов в задаче и ее связь с диаграммой коммуникаций.

(58) Диаграмма состояний на UML. Понятие состояния объекта.

(59) Паттерны выявления состояний.

(60) Специальные состояния – начальное и конечное. Супер состояние и его использование.

(61) Условия на диаграмме состояний.

(62) Правила перехода из состояния в состояния и из запись на диаграмме.

(63) Задача «Идентификация проектных элементов». Роль и выполняемые шаги.

(64) Понятие подсистемы и ее представление на UML.

(65) Подсистема и компонента.

(66) Понятие интерфейса и его представление на UML.

(67) Понятие пакета и правила их использование для структуризации визуальной модели.

(68) Зависимости пакетов и правила их выявления.

(69) Задача «Идентификация проектных механизмов». Роль и выполняемые шаги.

(70) Представление механизмов в визуальной модели.

(71) Задача «Проектирование сценария использования». Роль и выполняемые шаги.

(72) Задача «Проектирование подсистем». Роль и выполняемые шаги.

(73) Построить модель предметной области по постановке задачи.

(74) Найти сценарии использования ИС по постановке задачи.

(75) Построить модель анализа по заданному сценарию использования.

1. **РЕСУРСЫ**
	1. **Основная литература**
	2. Ларман К. Применение UML 2.0 и шаблонов проектирования. Введение в объектно-ориентированный анализ, проектирование и итеративную разработку. 3-е изд.: Пер. с англ.: -М.: Вильямс, 2013, 2016 (или более новое издание этой книги).
	3. Вигерс К., Битти Дж.: Разработка требований к программному обеспечению. Пер. с англ.: - М.: BHV, 2014, 2016, 2017, 2018 (или более новое издание этой книги).
	4. **Дополнительная литература**
2. Oussalah M. Software Architecture. John Wiley & Sons, 2014. - Режим доступа: <https://ebookcentral.proquest.com/lib/hselibrary-ebooks/detail.action?docID=1688022>
3. Chiang R., Siau K., Hardgrave B. Systems Analysis and Design: Techniques, Methodologies, Approaches, and Architecture. Routledge, 2009. - Режим доступа: <https://ebookcentral.proquest.com/lib/hselibrary-ebooks/detail.action?docID=435202>
4. Schmidt R., Schmidt R. F. Software Engineering: Architecture-Driven Software Development. Elsevier Science & Technology, 2013. - Режим доступа: <https://ebookcentral.proquest.com/lib/hselibrary-ebooks/detail.action?docID=1187150>
	1. **Программное обеспечение**

|  |  |  |
| --- | --- | --- |
| **№ п/п** | **Наименование** | **Условия доступа** |
|   1. | Microsoft Windows 10 | *Из внутренней сети университета (договор)* |
| 2. | Microsoft Office Professional Plus 2010 и новее | *Из внутренней сети университета (договор)* |
| 3. | CASE-средство Modelio версии 3 и новее | *Свободно распространяемое ПО с открытым исходным кодом (лицензия GPL): https://www.modelio.org/* |

* 1. **Профессиональные базы данных, информационные справочные системы,
	интернет-ресурсы (электронные образовательные ресурсы)**

|  |  |  |
| --- | --- | --- |
| **№ п/п** | **Наименование** | **Условия доступа** |
|  | ***Профессиональные базы данных, информационно-справочные системы*** |
| 1. | Электронно-библиотечная система Юрайт  | URL: https://biblio-online.ru/ |
|  | ***Интернет-ресурсы (электронные образовательные ресурсы)*** |
| 1. | Официальный сайт группы OMG с материалами по языку UML | URL: http://www.uml.org/ |
| 2 | Сайт с полной спецификацией языка UML | URL: https://www.omg.org/spec/UML |

* 1. **Материально-техническое обеспечение дисциплины**

Учебные аудитории для лекционных занятий по дисциплине обеспечивают использование и демонстрацию тематических иллюстраций, соответствующих программе дисциплины в составе:

ПЭВМ с доступом в Интернет (операционная система, офисные программы, антивирусные программы);

мультимедийный проектор с дистанционным управлением.

Учебные аудитории для лабораторных и самостоятельных занятий по дисциплине оснащены ПЭВМ, с возможностью подключения к сети Интернет и доступом к электронной информационно-образовательной среде НИУ ВШЭ.