

МИНИСТЕРСТВО ОБРАЗОВАНИЯ РЕСПУБЛИКИ БЕЛАРУСЬ

Учебно-методическое объединение по образованию
в области информатики и радиоэлектроники

УТВЕРЖДАЮ

Первый заместитель Министра образования
Республики Беларусь

_____ А.Г. Баханович

Регистрационный № _____

ПРОГРАММИРОВАНИЕ СЕТЕВЫХ ПРИЛОЖЕНИЙ

**Примерная учебная программа по учебной дисциплине
для специальности**

6-05-0611-01 Информационные системы и технологии

СОГЛАСОВАНО

Председатель Учебно-методического
объединения по образованию в
области информатики и
радиоэлектроники

_____ В.А. Богуш

СОГЛАСОВАНО

Начальник Главного управления
профессионального образования
Министерства образования
Республики Беларусь

_____ С.Н. Пищов

СОГЛАСОВАНО

Проректор по научно-методической
работе Государственного учреждения
образования «Республиканский
институт высшей школы»

_____ И.В. Титович

Эксперт-нормоконтролер

Минск 2024

СОСТАВИТЕЛИ:

В.Н.Комличенко, доцент кафедры экономической информатики учреждения образования «Белорусский государственный университет информатики и радиоэлектроники», кандидат технических наук, доцент;

Д.А.Сторожев, старший преподаватель кафедры экономической информатики учреждения образования «Белорусский государственный университет информатики и радиоэлектроники», магистр экономических наук, исследователь технических наук;

М.Н.Салапура, старший преподаватель кафедры экономической информатики учреждения образования «Белорусский государственный университет информатики и радиоэлектроники», магистр экономических наук;

Н.О.Петрович, старший преподаватель кафедры экономической информатики учреждения образования «Белорусский государственный университет информатики и радиоэлектроники», магистр экономических наук, исследователь технических наук.

РЕЦЕНЗЕНТЫ:

Кафедра дискретной математики и алгоритмики Белорусского государственного университета (протокол № 4 от 16.11.2023);

В.А.Грушев, заместитель директора по производству Иностранного унитарного научно-производственного предприятия «САМСОЛЮШНС», кандидат технических наук.

РЕКОМЕНДОВАНА К УТВЕРЖДЕНИЮ В КАЧЕСТВЕ ПРИМЕРНОЙ:

Кафедрой экономической информатики учреждения образования «Белорусский государственный университет информатики и радиоэлектроники» (протокол № 5 от 14.11.2023);

Научно-методическим советом учреждения образования «Белорусский государственный университет информатики и радиоэлектроники» (протокол №__ от _____);

Научно-методическим советом по прикладным информационным системам и технологиям Учебно-методического объединения по образованию в области информатики и радиоэлектроники (протокол № 5 от 11.12.2023)

ПОЯСНИТЕЛЬНАЯ ЗАПИСКА

ХАРАКТЕРИСТИКА УЧЕБНОЙ ДИСЦИПЛИНЫ

Примерная учебная программа по учебной дисциплине «Программирование сетевых приложений» разработана для студентов учреждений высшего образования, обучающихся по специальности 6-05-0611-01 «Информационные системы и технологии» в соответствии с требованиями образовательного стандарта общего высшего образования и примерного учебного плана вышеуказанной специальности.

Актуальность изучения учебной дисциплины «Программирование сетевых приложений» обоснована необходимостью подготовки специалистов, владеющих фундаментальными знаниями и практическими навыками в области проектирования и программирования сетевых приложений и их модулей при решении практических задач.

Воспитательное значение учебной дисциплины «Программирование сетевых приложений» заключается в формировании у обучающихся математической культуры и научного мировоззрения; развитии исследовательских умений, аналитических способностей, креативности, необходимых для решения научных и практических задач; развитии познавательных способностей и активности: творческой инициативы, самостоятельности, ответственности и организованности; формировании способностей к саморазвитию, самосовершенствованию и самореализации.

Изучение данной учебной дисциплины способствует созданию условий для формирования интеллектуально развитой личности обучающегося, которой присущи стремление к профессиональному совершенствованию, активному участию в экономической и социально-культурной жизни страны, гражданская ответственность и патриотизм.

ЦЕЛЬ, ЗАДАЧИ УЧЕБНОЙ ДИСЦИПЛИНЫ

Цель учебной дисциплины: теоретическая и практическая подготовка, обеспечивающая получение базовых знаний по созданию сетевых программ.

Задачи учебной дисциплины:

освоение возможностей, предоставляемых современными компьютерными сетевыми технологиями;

изучение принципов проектирования, создания, масштабирования сетевых приложений;

овладение методами, подходами, принципами, протоколами создания приложений, реализующих сетевое взаимодействие;

приобретение знаний и навыков проектирования и создания сетевых приложений;

освоение навыков программирования при создании сетевого взаимодействия между компонентами;

приобретение навыков работы в интегрированной среде разработки при создании сетевых приложений.

Базовыми учебными дисциплинами по курсу «Программирование сетевых приложений» являются «Иностранный язык», «Дискретная математика», «Основы алгоритмизации и программирования», «Объектно-ориентированное проектирование и программирование». В свою очередь, учебная дисциплина «Программирование сетевых приложений» является базой для таких учебных дисциплин компонента учреждения образования, как «Распределенные информационные системы» и «Программирование мобильных информационных систем». Содержательно учебная дисциплина «Программирование сетевых приложений» связана с такой учебной дисциплиной как «Средства и технологии анализа и разработки информационных систем».

ТРЕБОВАНИЯ К УРОВНЮ ОСВОЕНИЯ СОДЕРЖАНИЯ УЧЕБНОЙ ДИСЦИПЛИНЫ

В результате изучения учебной дисциплины «Программирование сетевых приложений» формируются следующие компетенции:

универсальные:

владеть основами исследовательской деятельности, осуществлять поиск, анализ и синтез информации;

решать стандартные задачи профессиональной деятельности на основе применения информационно-коммуникационных технологий;

быть способным к саморазвитию и совершенствованию в профессиональной деятельности;

проявлять инициативу и адаптироваться к изменениям в профессиональной деятельности;

базовая профессиональная: разрабатывать программные комплексы и системы для решения профессиональных задач на основе базовых технологий сетевого программирования, типовых решений, инструментальных и языковых средств создания приложений клиент-серверной архитектуры.

В результате изучения учебной дисциплины студент должен:

знать:

основы и особенности применяемого языка программирования при разработке клиент-серверных приложений;

методы разработки и отладки клиент-серверного приложения;

технологии и подходы к созданию клиент-серверного взаимодействия и разработке интерфейса пользователя;

уметь:

создавать приложения удаленного вызова процедур и методов, приложения в архитектуре клиент-сервер с различными методами и процедурами серверной обработки запросов клиентов;

разрабатывать приложения с различными типами клиентских приложений и интерфейсов, используя современные технологии;

использовать для разработки приложений наиболее распространенные сетевые протоколы обмена данными и другие средства передачи данных в клиент-серверных архитектурах;

владеть:

базовыми принципами и технологиями проектирования и программирования сетевых приложений;

техникой компонентно-ориентированной разработки клиент-серверных приложений;

методами разработки приложений распределенной обработки данных и технологии удаленных вызовов;

языками и техниками программирования серверных приложений и интерфейсов пользователя;

программными средствами разработки распределенных информационных систем;

инструментальными средствами разработки сетевых приложений.

Примерная учебная программа рассчитана на 120 учебных часов, из них – 68 аудиторных. Примерное распределение аудиторных часов по видам занятий: лекции – 28 часов, лабораторные занятия – 40 часов.

ПРИМЕРНЫЙ ТЕМАТИЧЕСКИЙ ПЛАН

Наименование раздела, темы	Всего аудиторных часов	Лекции	Лабораторные занятия
Раздел 1. Концепция и особенности объектно-ориентированного подхода при использовании языков программирования для разработки сетевых приложений	4	4	
Тема 1. Основные принципы, методы и перспективы разработки объектно-ориентированных программ и сетевых приложений на основе технологий Microsoft и Oracle	2	2	
Тема 2. Фундаментальные методы и свойства сетевой архитектуры, механизмы ее программной реализации в desktop/web-приложениях	2	2	
Раздел 2. Концептуальные основы языка программирования для разработки сетевых приложений	46	18	28
Тема 3. Обзор базовых конструкций и основных элементов языка	2	2	
Тема 4. Структурные элементы класса, методы взаимодействия объектов и организация наследования	6	2	4
Тема 5. Типы исключительных ситуаций и процесс их обработки	2	2	
Тема 6. Потoki ввода/вывода и работа с файлами	2	2	
Тема 7. Организация потоков, параллельной обработки, синхронизации и распределенной обработки синхронизируемых участков кода	6	2	4
Тема 8. Структурные механизмы языка программирования для реализации полиморфизма в программах	6	2	4
Тема 9. Средства языка для организации работы в сети. Основные классы и интерфейсы реализации сетевого взаимодействия	6	2	4
Тема 10. Библиотеки и средства внедрения визуальных компонент для организации GUI-интерфейсов пользователя. Обработка событий	6	2	4
Тема 11. Проектирование и разработка приложений в архитектуре клиент-сервер с организацией взаимодействия с базой данных	10	2	8

Наименование раздела, темы	Всего аудиторных часов	Лекции	Лабораторные занятия
Раздел 3. Технологии web- программирования и разработки сетевых приложений	18	6	12
Тема 12. Основы применения расширенных языков гипертекстовой разметки документов и разработки клиентских и серверных скриптов	6	2	4
Тема 13. Разработка web-приложений с организацией обработки клиентских запросов	10	2	8
Тема 14. Концепция распределенной обработки данных и технологии удаленной обработки данных	2	2	
Итого:	68	28	40

СОДЕРЖАНИЕ УЧЕБНОГО МАТЕРИАЛА

Раздел 1. КОНЦЕПЦИЯ И ОСОБЕННОСТИ ОБЪЕКТНО-ОРИЕНТИРОВАННОГО ПОДХОДА ПРИ ИСПОЛЬЗОВАНИИ ЯЗЫКОВ ПРОГРАММИРОВАНИЯ ДЛЯ РАЗРАБОТКИ СЕТЕВЫХ ПРИЛОЖЕНИЙ

Тема 1. ОСНОВНЫЕ ПРИНЦИПЫ, МЕТОДЫ И ПЕРСПЕКТИВЫ РАЗРАБОТКИ ОБЪЕКТНО-ОРИЕНТИРОВАННЫХ ПРОГРАММ И СЕТЕВЫХ ПРИЛОЖЕНИЙ НА ОСНОВЕ ТЕХНОЛОГИЙ MICROSOFT И ORACLE

Предмет курса и содержание учебной дисциплины, ее связь с другими учебными дисциплинами. Две парадигмы программирования. Основные направления в программировании. Возникновение объектно-ориентированного программирования (ООП). Базовые принципы ООП. Реализации основных концепций объектно-ориентированного программирования: полиморфизма, наследования в языках высокого уровня. Технология разработки программ на основе ООП. Обзор возможностей языков и технологий, разработанных компаниями Microsoft и Oracle.

Тема 2. ФУНДАМЕНТАЛЬНЫЕ МЕТОДЫ И СВОЙСТВА СЕТЕВОЙ АРХИТЕКТУРЫ, МЕХАНИЗМЫ ЕЕ ПРОГРАММНОЙ РЕАЛИЗАЦИИ В DESKTOP/WEB-ПРИЛОЖЕНИЯХ

Понятие о сетевой архитектуре. Общие представления о процессе передачи данных по сети. Понятие протокола и механизмы взаимодействия системы «клиент-сервер». Обзор общих принципов построения многоуровневых приложений. Работа с сетью на уровне сокетов и потоков. Способы доступа к ресурсам сети из программных приложений при помощи URL.

Раздел 2. КОНЦЕПТУАЛЬНЫЕ ОСНОВЫ ЯЗЫКА ПРОГРАММИРОВАНИЯ ДЛЯ РАЗРАБОТКИ СЕТЕВЫХ ПРИЛОЖЕНИЙ

Тема 3. ОБЗОР БАЗОВЫХ КОНСТРУКЦИЙ И ОСНОВНЫХ ЭЛЕМЕНТОВ ЯЗЫКА

Переменные и типы. Преобразование и приведение типов, расширение типов. Динамическая инициализация, область действия и время жизни переменных. Циклы и логика. Массивы и строки.

Тема 4. СТРУКТУРНЫЕ ЭЛЕМЕНТЫ КЛАССА, МЕТОДЫ ВЗАИМОДЕЙСТВИЯ ОБЪЕКТОВ И ОРГАНИЗАЦИЯ НАСЛЕДОВАНИЯ

Компонентные характеристики в определении класса. Доступ к переменным и методам. Конструкторы. Создание объектов класса и время жизни объекта. Перегрузка и переопределение методов. Наследование класса (наследование членов данных, их сокрытие, унаследованные методы). Создание многоуровневой иерархии. Переопределение методов и их применение. Динамическая диспетчеризация методов. Понятие и использование абстрактных классов.

Тема 5. ТИПЫ ИСКЛЮЧИТЕЛЬНЫХ СИТУАЦИЙ И ПРОЦЕСС ИХ ОБРАБОТКИ

Необходимость обработки исключительных ситуаций. Основные принципы обработки исключений. Типы исключений. Обработка исключительных ситуаций. Объекты исключительных ситуаций. Стандартные исключительные ситуации. Определение и порождение собственных исключительных ситуаций.

Тема 6. ПОТОКИ ВВОДА/ВЫВОДА И РАБОТА С ФАЙЛАМИ

Чтение консольного ввода. Чтение символов. Чтение строк. Запись консольного вывода. Чтение и запись файлов. Классы и интерфейсы ввода/вывода. Каталоги. Поточные классы. Байтовые потоки. Буферизированные байтовые потоки. Символьные потоки. Использование поточного ввода/вывода. Сериализация. Классы и интерфейсы потоков ввода/вывода. Преимущества потоков.

Тема 7. ОРГАНИЗАЦИЯ ПОТОКОВ, ПАРАЛЛЕЛЬНОЙ ОБРАБОТКИ, СИНХРОНИЗАЦИИ И РАСПРЕДЕЛЕННОЙ ОБРАБОТКИ СИНХРОНИЗУЕМЫХ УЧАСТКОВ КОДА

Общее представление о потоках. Создание, остановка и соединение потоков. Планирование потоков. Управление потоками. Синхронизация. Синхронизированные методы. Синхронизация блоков операторов. Тупики. Коммуникация между потоками.

Тема 8. СТРУКТУРНЫЕ МЕХАНИЗМЫ ЯЗЫКА ПРОГРАММИРОВАНИЯ ДЛЯ РЕАЛИЗАЦИИ ПОЛИМОРФИЗМА В ПРОГРАММАХ

Использование структурных механизмов (интерфейсов, механизма перегрузки функций, механизма виртуальных функций, механизма перегрузки операций языка) для реализации возможности создания множественных определений для операций и функций. Расширение интерфейсов для создания многоуровневой структурированной иерархии классов.

Тема 9. СРЕДСТВА ЯЗЫКА ДЛЯ ОРГАНИЗАЦИИ РАБОТЫ В СЕТИ. ОСНОВНЫЕ КЛАССЫ И ИНТЕРФЕЙСЫ РЕАЛИЗАЦИИ СЕТЕВОГО ВЗАИМОДЕЙСТВИЯ

Распределенная обработка данных. Основы работы в сети. Клиент-сервер. Proxy-сервера. Адресация Internet. Сетевые классы и интерфейсы. Обзор сокетов. Зарезервированные сокет. Сокеты TCP/IP клиентов. Сокеты TCP/IP серверов. Дейтаграммы. Использование URL. Основные классы и интерфейсы реализации сетевого взаимодействия.

Тема 10. БИБЛИОТЕКИ И СРЕДСТВА ВНЕДРЕНИЯ ВИЗУАЛЬНЫХ КОМПОНЕНТ ДЛЯ ОРГАНИЗАЦИИ GUI-ИНТЕРФЕЙСОВ ПОЛЬЗОВАТЕЛЯ. ОБРАБОТКА СОБЫТИЙ

Архитектура «Модель-Представление-Контроллер» (MVC). Создание графического интерфейса при помощи встроенных классов. Компоненты и

контейнеры. Использование элементов управления, менеджеров компоновки и меню. Элементы управления. Основные понятия. Добавление и удаление элементов управления. Реагирование на элементы управления. Понятие менеджера компоновки. Работа с меню и диалоговыми окнами.

Обработка событий. Модель делегирования событий. Источники событий. Блок прослушивания событий. Классы событий. Элементы-источники событий. Интерфейсы прослушивания событий. Использование модели делегирования событий.

Тема 11. ПРОЕКТИРОВАНИЕ И РАЗРАБОТКА ПРИЛОЖЕНИЙ В АРХИТЕКТУРЕ КЛИЕНТ-СЕРВЕР С ОРГАНИЗАЦИЕЙ ВЗАИМОДЕЙСТВИЯ С БАЗОЙ ДАННЫХ

Понятия и терминология ODBC-JDBC. Связь JDBC и ODBC. Настройка базы данных. DriverManager. Создание соединения с источником данных. URL и ODBC-JDBC. Более сложные соединения. Драйвер JDBC-ODBC Bridge. Получение метаданных для множества результатов.

Раздел 3. ТЕХНОЛОГИИ WEB-ПРОГРАММИРОВАНИЯ И РАЗРАБОТКИ СЕТЕВЫХ ПРИЛОЖЕНИЙ

Тема 12. ОСНОВЫ ПРИМЕНЕНИЯ РАСШИРЕННЫХ ЯЗЫКОВ ГИПЕРТЕКСТОВОЙ РАЗМЕТКИ ДОКУМЕНТОВ И РАЗРАБОТКИ КЛИЕНТСКИХ И СЕРВЕРНЫХ СКРИПТОВ

Введение в языки гипертекстовой разметки документов. Основные элементы и структура языков. Понятие и особенности разработки и использования клиентских и серверных скриптов. Запуск и выполнение клиентских и серверных скриптов. Способы организации и хранения данных и механизмы их представления.

Тема 13. РАЗРАБОТКА WEB-ПРИЛОЖЕНИЙ С ОРГАНИЗАЦИЕЙ ОБРАБОТКИ КЛИЕНТСКИХ ЗАПРОСОВ

Технологии расширения функциональных возможностей Web серверов. Обработка клиентского запроса: чтение параметров данных формы. Обработка клиентского запроса: заголовки HTTP-запроса. Генерация ответов сервера: коды состояния HTTP. Генерация ответов сервера: заголовки HTTP-ответов. Обработка cookies. Элементы сценариев web-страниц. Веб-ориентированные способы и модели структурирования и хранения данных, а также механизмы представления этой информации в браузере (клиенту).

Тема 14. КОНЦЕПЦИЯ РАСПРЕДЕЛЕННОЙ ОБРАБОТКИ ДАННЫХ И ТЕХНОЛОГИИ УДАЛЕННОЙ ОБРАБОТКИ ДАННЫХ

Понятие и архитектура распределенной системы. Протоколы и программная реализация удаленного вызова процедур. Объектно-ориентированные вызовы удаленных методов. Архитектура решений, основанных на Web Services. Протоколы и стандарты. Публикация и развертывание служб.

ИНФОРМАЦИОННО-МЕТОДИЧЕСКАЯ ЧАСТЬ**ЛИТЕРАТУРА****ОСНОВНАЯ**

1. Лой, М. Програмуем на Java / М. Лой, П. Нимайер, Д. Лук. – 5-е изд. – Санкт-Петербург: Питер, 2023. – 544 с.
2. Макгрегор, Д. От Java к Kotlin / Д. Макгрегор, Н. Прайс ; пер. с англ. – Санкт-Петербург : БХВ-Петербург, 2023. – 448 с.
3. Шилдт, Г. Полное руководство. Java / Г. Шилдт. – 12-е изд. – Москва : Диалектика, 2022. – 1344 с.
4. Парлог, Н. Система модулей Java / Н. Парлог ; пер. с англ. А. Павлова. – Санкт-Петербург : Питер, 2021. – 464 с.
5. Смолякова, О. Г. Основы разработки веб-приложений на языке программирования Java : учебно-методическое пособие / О. Г. Смолякова. – Минск : БГУИР, 2019. – 131 с.
6. Java Persistence API и Hibernate / К. Бауэр, Г. Кинг, Г. Грегори ; пер. с англ. Д.А. Зинкевича. – Москва : ДМК Пресс, 2017. – 632 с.
7. Эккель, Б. Философия Java / Б. Эккель. – 4-е полное изд. – Санкт-Петербург : Питер, 2018. – 1168 с.
8. Лафоре, Р. Структуры данных и алгоритмы Java / Р. Лафоре. – 2-е изд. – Санкт-Петербург : Питер, 2018. – 704 с.
9. Седжвик, Р. Computer Science : основы программирования на Java, ООП, алгоритмы и структуры данных / Р. Седжвик. – Санкт-Петербург : Питер, 2018. – 1072 с.
10. Гонсалвес, Э. Изучаем Java EE 7 / Э. Гонсалвес. – Санкт-Петербург : Питер, 2017. – 640 с.
11. Курняван, Б. Программирование web-приложений на языке Java / Б. Курняван. – Москва : Лори, 2014. – 880 с.
12. Васильев, А. Н. Java. Объектно-ориентированное программирование : для магистров и бакалавров / А. Н. Васильев. – Санкт-Петербург : Питер, 2014. – 400 с.
13. Гупта, А. Java EE 7. Основы / А. Гупта ; пер. с англ. А. Г. Гузикевича. – Москва : Вильямс, 2014. – 336 с.
14. Герман, О. В. JAVA и Интернет-бизнес / О. В. Герман, Ю. О. Герман. – Минск : Бестпринт, 2010. – 384 с.
15. Holzner, S. JAVA 2 : Programming ; Black Book / S. Holzner. – New Delhi : Paraglyph Press, 2010. – 1460 p.
16. Эккель, Б. Философия Java / Б. Эккель. – 4-е изд. – Санкт-Петербург : Питер, 2009. – 640 с.
17. Хемраджани, А. Гибкая разработка приложений на Java с помощью Spring, Hibernate и Eclipse / А. Хемраджани ; пер. с англ. – Москва : Вильямс, 2008. – 352 с.
18. Хорстманн, К. С. Java 2 : в 2 т. / К. С. Хорстманн, Г. Корнелл. – 8-е изд. – Москва : Вильямс, 2008. – Т. 1 Основы. – 816 с.

19. Хорстманн, К. С. Java 2 : в 2 т. / К. С. Хорстманн, Г. Корнелл. – 8-е изд. – Москва : Вильямс, 2009. – о Т. 2 : Тонкости программирования. – 992 с.
20. Герман, О. В. Программирование на Java и C# для студента / О. В. Герман, Ю. О. Герман. – Санкт-Петербург : БХВ-Петербург, 2005. – 512 с.
21. Галиаскаров, Э. Г. Анализ и проектирование систем с использованием UML : учебное пособие / Э. Г. Галиаскаров, А. С. Воробьев. – Москва : Юрайт, 2022. – 125 с.
22. Олифер, В. Г. Компьютерные сети : принципы, технологии, протоколы / В. Г. Олифер, Н. А. Олифер. – Санкт-Петербург : Питер, 2021. – 1008 с.
23. Современные сетевые технологии : учебно-методическое пособие / В. А. Леванцевич [и др.]. – Минск : БГУИР, 2020. – 120 с.
24. Букатов, А. А. Компьютерные сети : расширенный начальный курс : учебное пособие / А. А. Букатов, С. А. Гуда. – Санкт-Петербург : Питер, 2020. – 496 с.
25. Умрихин, Е. Разработка веб-приложений с помощью ASP.Net Core MVC / Е. Умрихин. – Санкт-Петербург : БХВ-Петербург, 2023. – 416 с.
26. Прайс, М. C# 10 и .NET 6. Современная кросс-платформенная разработка / М. Прайс. – 6-е изд. – Санкт-Петербург : Питер, 2023. – 848 с.
27. Тепляков, С. Паттерны проектирования на платформе .NET / С. Тепляков. – Санкт-Петербург : Питер, 2021. – 320 с.
28. Рихтер, Д. CLR via C#. Программирование на платформе Microsoft .NET Framework 4.5 на языке C# / Д. Рихтер. – 4-е изд. – Санкт-Петербург : Питер, 2019. – 896 с.
29. Троелсен, Э. Язык программирования C# 5.0 и платформа .NET 4.5 / Э. Троелсен. – 6-е изд. – Москва : Вильямс, 2016. – 1312 с.
30. .NET. Сетевое программирование / В. Кумар [и др.]. – Москва : Лори, 2014. – 400 с.
31. Перри, Б. У. Java сервлеты и JSP сборник рецептов / Б. У. Перри. – 2-е изд. – Москва : Кудиц-образ, 2006. – 768 с.
32. Мориссо-Леруа, Н. ORACLE8i Java-компонентное программирование при помощи EJB, COBRA и JSP / Н. Мориссо-Леруа. – Москва : Лори, 2002. – 484 с.
33. Фленов, М. Е. C# глазами хакера / М. Е. Фленов. – Санкт-Петербург : БХВ-Петербург, 2023. – 224 с.
34. Тузовский, А. Ф. Проектирование и разработка Web-приложений : учебное пособие / А. Ф. Тузовский. – Москва : Юрайт, 2021. – 218 с.
35. Сеттер, Р. В. Изучаем Java на примерах и задачах / Р. В. Сеттер. – Санкт-Петербург : Наука и техника, 2016. – 240 с.
36. Флэнаган, Д. Java в примерах : справочник / Д. Флэнаган. – Санкт-Петербург : Символ, 2016. – 664 с.
37. Нимейер, П. Программирование на Java / П. Нимейер, Д. Леук. – Москва : Эксмо, 2014. – 1312 с.
38. Портянкин, И. Swing. Эффективные пользовательские интерфейсы / И. Портянкин. – Москва : Лори, 2011. – 608 с.
39. Прохоренок, Н. JavaFX / Н. Прохоренок. – Санкт-Петербург : БХВ-Петербург, 2020. – 768 с.

40. Куликов, С. С. Работа с MySQL, MS SQL Server и Oracle в примерах : учебно-методическое пособие : в 2 ч. / С. С. Куликов, Е. Е. Фадеева. – Минск : БГУИР, 2019. – Ч. 1. – 287 с.

41. Садаладж, П. Д. NoSQL : новая методология разработки нереляционных баз данных / П. Д. Садаладж, М. Фаулер. – Москва : Вильямс, 2017. – 192 с.

ДОПОЛНИТЕЛЬНАЯ

42. Лафоре, Р. Структуры данных и алгоритмы в Java / Р. Лафоре. – 2-е изд. – Санкт-Петербург : Питер, 2015. – 704 с.

43. Блох, Д. Java. Эффективное программирование / Д. Блох. – Москва : Лори, 2016. – 464 с.

44. Мартин, Р. Чистый код. Создание, анализ и рефакторинг / Р. Мартин. – Санкт-Петербург : Питер, 2015. – 464 с.

45. Эванс, Б. Java. Новое поколение разработки / Б. Эванс, М. Вербург. – Санкт-Петербург : Питер, 2014. – 560 с.

46. Урбэртон, Р. Лямбда-выражения в Java 8. Функциональное программирование – в массы / Р. Урбэртон. – Москва : ДМК Пресс, 2014. – 192 с.

47. Лонг, Ф. Руководство для программиста на Java: 75 рекомендаций по написанию надежных и защищенных программ / Ф. Лонг [и др.]. – Москва : Вильямс, 2014. – 256 с.

48. Машнин, Т. Java FX 2.0. Разработка RIA-приложений / Т. Машнин. – Санкт-Петербург : БХВ-Петербург, 2012. – 320 с.

49. Буч, Г. Введение в UML от создателей языка / Г. Буч, Д. Рамбо, И. Якобсон. – Москва : ДМК Пресс, 2015. – 780 с.

50. Гамма, Э. Приемы объектно-ориентированного проектирования. Паттерны проектирования / Э. Гамма, Р. Хелм, Р. Джонсон [и др.]. – Санкт-Петербург : Питер, 2015. – 1087 с.

51. Чакон, С. Git для профессионального программиста / С. Чакон, Б. Страуб. – Санкт-Петербург : Питер, 2016. – 496 с.

52. Крейг, У. Spring в действии / У. Крейг. – Москва : ДМК, 2015. – 752 с.

53. Мак-Лахлин, Б. Java и XML / Б. Мак-Лахлин ; пер. с англ. – Санкт-Петербург : Символ, 2015. – 544 с.

МЕТОДИЧЕСКИЕ РЕКОМЕНДАЦИИ ПО ОРГАНИЗАЦИИ И ВЫПОЛНЕНИЮ САМОСТОЯТЕЛЬНОЙ РАБОТЫ СТУДЕНТОВ

При изучении учебной дисциплины рекомендуется использовать следующие формы самостоятельной работы:

контролируемая самостоятельная работа в виде решения индивидуальных заданий в аудитории во время проведения лабораторных занятий под контролем преподавателя;

работа с основной и дополнительной литературой, интернет-ресурсами.

ПЕРЕЧЕНЬ РЕКОМЕНДУЕМЫХ СРЕДСТВ ДИАГНОСТИКИ КОМПЕТЕНЦИЙ СТУДЕНТА

Примерным учебным планом по специальности 6-05-0611-01 «Информационные системы и технологии» в качестве формы промежуточной аттестации по учебной дисциплине «Программирование сетевых приложений» рекомендуется экзамен и курсовой проект. Оценка учебных достижений студента производится по десятибалльной шкале.

Для текущего контроля по учебной дисциплине и диагностики компетенций студентов могут использоваться следующие формы:

- отчеты по лабораторным работам с их устной защитой;
- защита индивидуальной практической работы;
- защита контрольной работы;
- устный опрос;
- электронные тесты;
- коллоквиум;
- оценивание на основе модульно-рейтинговой системы.

РЕКОМЕНДУЕМЫЕ МЕТОДЫ (ТЕХНОЛОГИИ) ОБУЧЕНИЯ

Основные рекомендуемые методы (технологии) обучения, отвечающие целям и задачам учебной дисциплины:

- проблемное обучение (проблемное изложение, вариативное изложение, частично-поисковый метод), реализуемое на лекционных занятиях;
- устное изложение учебного материала на лекциях, сопровождаемое показом и демонстрацией слайдов, видеоматериалов;
- учебно-исследовательская деятельность, творческий подход, реализуемые на лабораторных занятиях.

РЕКОМЕНДАЦИИ ПО КУРСОВОМУ ПРОЕКТИРОВАНИЮ

Цель курсового проекта: освоение практических навыков проектирования и создания приложения в архитектуре клиент-сервер с организацией взаимодействия с базой данных.

Курсовой проект представляет собой решение по проектированию и программированию программного обеспечения с пользовательским интерфейсом, реализующим сетевое взаимодействие.

Курсовой проект выполняется индивидуально. В проекте студент должен продемонстрировать умение применять все знания, которые были получены в ходе лекционных и лабораторных занятий.

При выполнении курсового проекта может быть предложен выбор 2 архитектур программного средства.

Архитектура №1.

Общие требования: приложение должно быть выполнено в архитектуре клиент-сервер с многопоточным сервером с организацией взаимодействия с базой данных на объектно-ориентированном языке Java или C#.

Уровни архитектуры: серверное приложение может быть реализовано в виде консольного приложения или GUI-приложения. Настройки сервера должны меняться без изменения исходного кода (аргументы командной строки, конфиг-файлы и т.д.).

Клиентское приложение: оконное приложение с использованием стандартных библиотек пользовательского интерфейса (WCF, WPF, JavaFX, SWING, AWT).

В рамках работы над курсовым проектом должны быть использованы следующие техники:

1) разработка и использование собственной иерархии классов (не менее 5), расширение базовых классов, предоставляемых JDK/SDK;

2) реализация не менее 2-х паттернов проектирования на свой выбор (по желанию);

3) использовать сокрытие данных (инкапсуляция), перегрузку методов, переопределение методов, сериализацию, абстрактные типы данных (интерфейсы, абстрактные классы), статические методы, обработку исключительных ситуаций.

Бизнес-логика: бизнес-логика системы должна быть реализована только на серверной части приложения. На сервере должна быть предусмотрена возможность параллельной обработки запросов. Пользовательский интерфейс должен обеспечивать доступ ко всем функциям. Функциональные возможности серверной части должны насчитывать не менее 12 высокоуровневых вариантов использования, исключая тривиальные операции работы с БД (добавление, удаление, редактирование записей в БД). Необходимо предусмотреть механизм авторизации всех пользователей. Количество ролей – не менее 2-3 в зависимости от специфики предметной области (администратор, сотрудник компании, гость). Проектирование и разработку реляционной модели (не менее 5 связанных таблиц) предметной области осуществить с использованием процедуры нормализации отношений (до 3 НФ). Связи между таблицами должны быть определены как идентифицирующие и не идентифицирующие, с указанием мощности связи. Схема базы данных должна обеспечивать ссылочную целостность данных. Доступ к данным в системе управления базами данных (СУБД) должен осуществляться через драйвер, предоставляемый производителем СУБД или через использование специальных технологий.

Конкретные версии фреймворков и технологий, применяемых для реализации программного средства, должны быть актуальными на начало учебного года.

Обязательные требования к функционалу системы:

1) в разрабатываемом приложении обеспечить добавление, редактирование и удаление записей из базы данных, сохранение табличных результатов в файле (создание текстового отчета), предоставление пользователю аналитической информации (графики, диаграммы);

2) предусмотреть возможность сохранять информацию в любой момент на сервере и загружать ранее сохраненные данные;

3) предусмотреть механизм авторизации пользователей (роли Администратор и Пользователь).

База данных (не менее пяти связанных таблиц) должна быть приведена к 3-ей нормальной форме.

Архитектура №2.

Общие требования: программное средство следует разработать в архитектуре: web-приложение с базой данных с использованием объектно-ориентированного языка программирования Java или C# (ASP.NET), современных технологий и фреймворков. Конкретные версии фреймворков и технологий, применяемых для реализации программного средства, должны быть актуальными на начало учебного года. В рамках работы должны быть представлены: разработка и использование собственной иерархии классов, реализация не менее 2-х паттернов проектирования.

Архитектурная модель: сервер СУБД – СУБД для размещения базы данных курсового проекта. Сервер Приложений – используется для размещения «серверной» части приложения, представленной Моделью на основе ORM технологии (Hibernate/JPA), Бизнес-логика приложения (серверная часть) должна быть реализована на основе фреймворка Spring или EJB/Servlets. Клиентская часть может быть реализована на веб-технологиях: JS/JSP/JSF/React/Angular. Бизнес-логику необходимо реализовать только на серверной части. На сервере следует предусмотреть возможность параллельной обработки запросов. СУБД для создания базы данных реляционной или нереляционной. Проектирование и разработку реляционной модели (не менее 5 связанных таблиц) предметной области осуществить с использованием процедуры нормализации отношений (до 3 НФ). Связи между таблицами должны быть определены как идентифицирующие и не идентифицирующие, с указанием мощности связи. Схема базы данных должна обеспечивать ссылочную целостность данных.

Функциональность: пользовательский интерфейс должен обеспечивать доступ ко всем функциям. Функциональные возможности серверной части должны насчитывать не менее 12 высокоуровневых вариантов использования, исключая тривиальные операции работы с БД (добавление, удаление, редактирование записей в БД). Необходимо предусмотреть механизм авторизации всех пользователей. Количество ролей – не менее 2-3 в зависимости от специфики предметной области (администратор, сотрудник компании, гость). Приложение должно задействовать функционал стороннего сервиса, представленного в виде публичного REST API (Google YouTube, Maps, Translate, Drive). База данных должна быть приведена к 3-ей нормальной форме.

Обеспечивающая инфраструктура : программное средство должно выполняться в операционной системе Windows 8 и выше с возможной предустановкой библиотек или пакетов выбранной среды программирования. Программная документация представляется в составе руководства по установке

(развертыванию) программного средства и руководства пользователя для всех категорий.

Требования к информационной безопасности : организовать процесс защиты хранимых и передаваемых данных (шифрование, резервное копирование данных).

Требования к поставке : разработанное приложение поставляется в виде 2-х либо 3-х Docker-контейнеров (опционально), инструкции и набора интеграционных тестов: контейнер с базой данных, контейнер с сервером приложений, контейнер с клиентским приложением, проект интеграционных тестов (не менее 20 тестов), показывающих работоспособность сервера приложений, инструкция с описанием процесса развертывания приложения (часть пояснительной записки к проекту).

Рекомендуемое содержание пояснительной записки (перечень подлежащих разработке вопросов):

Титульный лист. Реферат. Задание по курсовому проекту. Содержание. Перечень условных обозначений, символов и терминов.

Введение.

1. Анализ и моделирование предметной области программного средства.
1.1. Описание предметной области. 1.2. Разработка функциональной модели предметной области. 1.3. Анализ требований к разрабатываемому программному средству. Спецификация функциональных требований. 1.4. Разработка информационной модели предметной области. 1.5. UML-модели представления программного средства и их описание.

2. Проектирование и конструирование программного средства.
2.1. Постановка задачи. 2.2. Архитектурные решения. 2.3. Описание алгоритмов, реализующих ключевую бизнес-логику разрабатываемого программного средства. 2.4. Проектирование пользовательского интерфейса. 2.5. Обоснование выбора компонентов и технологий для реализации программного средства.

3. Тестирование и проверка работоспособности программного средства.

4. Инструкция по развертыванию приложения и сквозной тестовый пример, начиная от авторизации, демонстрируя реализацию всех вариантов использования.

Заключение. Список использованных источников. Приложения (обязательные): отчет о проверке на заимствования в системе «Антиплагиат»; листинг кода алгоритмов, реализующих основную бизнес-логику; листинг скрипта генерации базы данных. Ведомость документов курсового проекта.

Перечень графического материала (с указанием обязательных чертежей и графиков):

1. IDEF0/BPMN-модель процессов предметной области (чертеж, 1 лист формата А1-А4).

2. Схема алгоритма, реализующая основную бизнес-логику программного средства (чертеж, 1 лист формата А1-А4).

3. Плакаты, отражающие результаты проектирования программного средства (3 листа формата А1-А4):

- 3.1. UML диаграмма классов (плакат, 1 лист формата А1-А4).
- 3.2. Модели представления программного средства (плакат, 1 лист формата А1-А4).
- 3.3. Скриншоты рабочих окон программного средства (плакат, 1 лист формата А1-А4).

ПРИМЕРНЫЙ ПЕРЕЧЕНЬ ТЕМ КУРСОВЫХ ПРОЕКТОВ

1. Разработка системы экстраполяции показателей объемов продаж на предприятии.
2. Разработка автоматизированной системы оценки финансовых рисков компании.
3. Разработка системы расчета коэффициентов финансовой устойчивости (показателей структуры капитала).
4. Разработка системы учета предоставления правовой информации на предприятии.
5. Разработка системы прогнозирования объемов продаж на основе трендовых моделей.
6. Разработка системы учета затрат и калькуляции себестоимости с использованием нормативных затрат (модель Стандарт Костинг).
7. Разработка автоматизированной системы бюджетирования на предприятии.
8. Разработка системы рейтинговой оценки предприятия на основе полученных значений финансовых коэффициентов.
9. Разработка системы для расчета эффективных процентных ставок по кредитам.
10. Разработка системы оценки деятельности субъекта хозяйствования на основе показателей рентабельности.
11. Разработка системы оценки уровня качества продукции.
12. Разработка системы прогнозирования продаж на основе мультипликативной модели.
13. Разработка системы автоматизации работы банка с физическими лицами (выбрать любое направление и согласовать с руководителем).
14. Разработка системы дисконтирования и оценки стоимости капитала.
15. Разработка системы автоматизации работы аэропорта.
16. Разработка системы оценки стоимости компании на основе модели Эдвардса-Белла-Ольсона.
17. Разработка системы автоматизации оценки этапов выполнения курсового проектирования.
18. Разработка системы управления бюджетом на предприятии.
19. Разработка системы моделирования вероятности банкротства организаций.
20. Разработка системы дисконтирования и оценки стоимости капитала.
21. Разработка системы экстраполяции показателей объемов продаж товаров (услуг).

22. Разработка системы оценки качества и конкурентоспособности продукции предприятия.

23. Разработка системы оценки финансовых рисков активов и пассивов бухгалтерского баланса предприятия.

24. Разработка системы оценки эффективности инвестиционного проекта.

25. Разработка системы оценки конкурентоспособности выпускаемой продукции.

26. Разработка системы автоматизации розничного бизнеса банка.

27. Разработка автоматизированной системы анализа прибыльности и оценки ликвидности компании.

28. Разработка системы анализа структуры капитала и долгосрочной платежеспособности.

29. Разработка системы автоматизации оценки объектов недвижимости.

30. Система анализа инвестиционной привлекательности организации-эмитента.

31. Система оценки стоимости объектов недвижимости на основе затратного подхода.

32. Разработка автоматизированной системы анализа затрат по методике Pre ABC-Costing.

33. Разработка автоматизированной системы оценки стоимости компании на основе модели Ольсона.

34. Разработка автоматизированной системы оценки рисков при управлении проектами.

35. Разработка системы прогнозирования финансовой устойчивости предприятия на основе модели Янсона.

36. Разработка автоматизированной системы по расчету студенческих стипендий.

37. Разработка системы прогнозирования платежеспособности предприятия на основе модели Фулмера.

38. Разработка системы оценки качества продукции предприятия.

39. Разработка системы оценки стоимости компании на основе концепции экономической добавленной стоимости.

ПРИМЕРНЫЙ ПЕРЕЧЕНЬ ТЕМ ЛАБОРАТОРНЫХ ЗАНЯТИЙ

1. Разработка консольных приложений в инструментальной среде программирования. Использование классов и объектов с организацией наследования, работы с файлами и обработки исключительных ситуаций.

2. Разработка однопоточных сетевых приложений;

3. Разработка многопоточных сетевых приложений;

4. Разработка клиент-серверных приложений, взаимодействующих по протоколу TCP;

5. Разработка клиент-серверных приложений, взаимодействующих по протоколу UDP;

6. Разработка пользовательского интерфейса. Работа с фреймами, элементами управления, графикой. Создание соединения с БД и обработка пользовательских запросов.

7. Программирование клиент-серверных приложений, взаимодействующих с базой данных;

8. Применение языков расширенной разметки для эффективной организации структуры хранения информации и её визуализация на основе различных моделей представления данных;

9. Разработка простейших серверных WEB-приложений;

10. Создание приложений для распределенной обработки данных. Удаленный вызов методов. Удаленные интерфейсы. Реализация удаленного интерфейса. Создание заглушек и каркасов. Использование удаленного объекта.

ПРИМЕРНЫЙ ПЕРЕЧЕНЬ КОМПЬЮТЕРНЫХ ПРОГРАММ (необходимого оборудования, наглядных пособий и т. п.)

1. Операционная система Microsoft Windows 7 и выше.
2. Комплект разработчика приложений JDK версии 8 и выше.
3. Комплект разработчика приложений SDK.
4. Интегрированная среда разработки на объектно-ориентированном языке программирования IntelliJ IDEA или аналоги.
5. Интегрированная среда разработки на объектно-ориентированном языке программирования MS Visual Studio или аналоги.
6. Система управления базами данных MySQL или аналоги.
7. Браузер Google Chrome или аналоги.
8. CASE-инструмент для проектирования и конструирования программного обеспечения Enterprise Architect 7.1 и выше или аналоги.
9. Web-сервера Tomcat, JBoss and Glassfish.
10. Инструменты для сборки приложений: Gradle, Maven и Ant.
11. JaCoCo – инструмент для оценки охвата кода юнит тестами.
12. JMeter – инструмент с открытым исходным кодом для тестирования производительности приложений.
13. JUnit – библиотека для модульного тестирования программного обеспечения на Java.
14. Spark – веб-фреймворк для Java, предназначенный для быстрой разработки с минимальными накладными расходами.
15. Фреймворк Spring – инструмент для инъекций зависимостей.