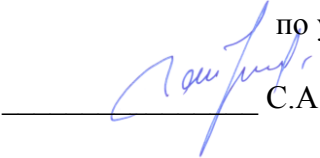


УТВЕРЖДАЮ
Зам. директора ИК
по учебной работе


С.А. Гайворонский

« 1 » 06 2016 г.

РАБОЧАЯ ПРОГРАММА ДИСЦИПЛИНЫ

ПРОЕКТИРОВАНИЕ ПРОГРАММНЫХ ПРИЛОЖЕНИЙ

На учебный год 2016-2017

НАПРАВЛЕНИЕ ООП **09.04.01 Информатика и вычислительная техника**

ПРОФИЛЬ ПОДГОТОВКИ **Информационное и программное обеспечение систем управления**

КВАЛИФИКАЦИЯ (СТЕПЕНЬ) **магистр**
БАЗОВЫЙ УЧЕБНЫЙ ПЛАН ПРИЕМА **2016 г.**
КУРС 2 СЕМЕСТР 3

КОЛИЧЕСТВО КРЕДИТОВ **6 кредитов**
КОД ДИСЦИПЛИНЫ **В.М3.3**

Виды учебной деятельности	Временной ресурс по очной форме обучения
Лекции, ч.	16
Лабораторные занятия, ч.	32
Аудиторные занятия, ч.	48
Самостоятельная работа, ч.	168
ИТОГО, ч.	216

ФОРМА ОБУЧЕНИЯ **очная**

ВИД ПРОМЕЖУТОЧНОЙ АТТЕСТАЦИИ **экзамен**
ОБЕСПЕЧИВАЮЩЕЕ ПОДРАЗДЕЛЕНИЕ **кафедра АиКС**

ЗАВЕДУЮЩИЙ КАФЕДРОЙ



Фадеев А.С.

РУКОВОДИТЕЛЬ ООП



Ким В.Л.

ПРЕПОДАВАТЕЛЬ



Баночкин П.И.

2016 г.

1. ЦЕЛИ ОСВОЕНИЯ ДИСЦИПЛИНЫ

Цель преподавания дисциплины «Проектирование программных приложений» заключается в формировании у студентов знаний, умений и навыков, теоретических и практических в области проектирования программных приложений в соответствии с принципами сервисно-ориентированной архитектуры приложений, в формировании у студентов мотивации к самообразованию за счет активизации самостоятельной познавательной деятельности. Поставленные цели полностью соответствуют целям (Ц1-Ц5) ООП.

2. МЕСТО ДИСЦИПЛИНЫ В СТРУКТУРЕ ООП

Дисциплина «Проектирование программных приложений» (М2.В.6.3) входит в состав вариативной части профессионального цикла дисциплин учебного плана (М2.В).

Для её успешного освоения необходимы базовые и специальные знания, полученные при изучении ООП бакалаврской подготовки по направлению «Информатика и вычислительная техника» и близких направлений.

В рамках ООП «Информатика и вычислительная техника» данной дисциплине предшествуют знания, полученные при изучении дисциплин **ПЕРЕКВИЗИТОВ**:

«Технология разработки программного обеспечения» (Б.М3).

КОРЕКВИЗИТЫ – «Технология разработки пользовательских интерфейсов» (В.М3.2).

3. РЕЗУЛЬТАТЫ ОСВОЕНИЯ ДИСЦИПЛИНЫ

В задачи изучения дисциплины входит изложение основных положений OLAP технологии баз данных, их применения при реализации промышленных систем.

В соответствии с требованием ООП освоение дисциплины направлено на формирование у студентов следующих компетенций (результатов обучения), в т. ч. в соответствии с ФГОС (табл. 1).

Таблица 1

Составляющие результатов обучения, которые будут получены при изучении данной дисциплины

Результаты обучения (компетенции из ФГОС)	Составляющие результатов обучения					
	Код	Знания	Код	Умения	Код	Владение опытом
Р4 (ОК-5, ПК-10, ПК-11, ПК-12,	35.4.4	Теоретических и практических	У5.4.4	Проектировать программные приложения;	В5.4.4	Методами проектирования программных

ПК-19, ОПК-1, ОПК-2, ОПК-3, ОПК-5, ОПК-6, ПК-13, ПК-14)		принципов организации программных приложений; способов организации веб- сервисов и основных протоколов передачи данных.		организовывать веб-сервисы и использовать необходимые протоколы передачи данных; использовать подход сервисно-ориентированной архитектуры для бизнес-задач предприятия.		приложений; приемами и основными техническими средствами организации веб-сервисов.
---	--	---	--	---	--	--

В результате освоения дисциплины «Методы проектирования систем логического управления» студентами должны быть достигнуты следующие результаты (табл. 2):

Таблица 2

Планируемые результаты освоения дисциплины

№ п/п	Результат
РД1	Знание теоретических и практических принципов организации программных приложений; способов организации веб- сервисов и основных протоколов передачи данных.
РД2	Уметь проектировать программные приложения; организовывать веб-сервисы и использовать необходимые протоколы передачи данных; использовать подход сервисно-ориентированной архитектуры для бизнес-задач предприятия.
РД3	Владеть методами проектирования программных приложений; приемами и основными техническими средствами организации веб-сервисов

В процессе освоения дисциплины у студентов развиваются следующие **компетенции:**

1. Универсальные (общекультурные) – использованием на практике умений и навыков в организации исследовательских и проектных работ, в управлении коллективом (ОК-5);

2. Общепрофессиональные:

способностью воспринимать математические, естественнонаучные, социально-экономические и профессиональные знания, умением самостоятельно приобретать, развивать и применять их для решения нестандартных задач, в том числе в новой или незнакомой среде и в междисциплинарном контексте (ОПК-1);

культурой мышления, способностью выстраивать логику рассуждений и высказываний, основанных на интерпретации данных, интегрированных из разных областей науки и техники, выносить суждения на основании неполных данных (ОПК-2);

способностью анализировать и оценивать уровни своих компетенций в сочетании со способностью и готовностью к саморегулированию дальнейшего

образования и профессиональной мобильности (ОПК-3);

владением методами и средствами получения, хранения, переработки и трансляции информации посредством современных компьютерных технологий, в том числе в глобальных компьютерных сетях (ОПК-5);

способностью анализировать профессиональную информацию, выделять в ней главное, структурировать, оформлять и представлять в виде аналитических обзоров с обоснованными выводами и рекомендациями (ОПК-6).

3. Профессиональные:

- способностью разрабатывать и реализовывать планы информатизации предприятий и их подразделений на основе Web- и CALS-технологий (ПК-10);
- способностью формировать технические задания и участвовать в разработке аппаратных и (или) программных средств вычислительной техники (ПК-11);
- способностью выбирать методы и разрабатывать алгоритмы решения задач управления и проектирования объектов автоматизации (ПК-12);
- способностью к программной реализации распределенных информационных систем (ПК-13);
- способностью к программной реализации систем с параллельной обработкой данных и высокопроизводительных систем (ПК-14);
- способностью к применению современных технологий разработки программных комплексов с использованием CASE-средств, контролировать качество разрабатываемых программных продуктов (ПК-19).

4. СТРУКТУРА И СОДЕРЖАНИЕ ДИСЦИПЛИНЫ

4.1. Аннотированное содержание разделов дисциплины

Раздел 1. Современные тенденции в области проектирования программных приложений. Архитектурные стили. Проектирование в жизненном цикле разработки программного обеспечения.

В разделе рассматриваются современные подходы к проектированию программных приложений, в том числе с использованием облачных технологий. Сравниваются длительность и степень формализации в различных моделях разработки программного обеспечения. Приводятся сценарии использования различных архитектурных стилей.

Раздел 2. Способы документирования проекта программного продукта.

В данном разделе излагаются способы представления проекта программного продукта с использованием стандартов Software Design Document и стандартов ГОСТ 19 серии. Рассматриваются способы представления архитектуры «4+1 Design Views» и «Views and beyond».

Перечень лабораторных работ по разделу:

1. Проектирование модульной и компонентной архитектуры программного обеспечения.

Раздел 3. Проектирование модели предметной области

Данный раздел посвящен вопросам проектирования модели предметной области и ее реализации. Рассматриваются подход Domain Driven Design и такие понятия, как сущность, ценный объект, репозиторий, сервис, агрегат. Приводятся рекомендации по созданию модели предметной области.

Перечень лабораторных работ по разделу:

1. Создание и реализация модели предметной области

Раздел 4. Архитектура веб-приложений

В данном разделе рассматриваются современные архитектурные решения для реализации веб-приложений: архитектурный шаблон MVVM, технологии Web-Socket и WebCL.

Перечень лабораторных работ по разделу:

1. Реализация MVVM веб-приложения.
2. Реализация WebCL приложения

Раздел 5. Сервис-ориентированная архитектура

В данном разделе рассматривается способ создания систем программных приложений с использованием сервис-ориентированного архитектурного стиля. Описываются средства создания REST и SOAP веб-сервисов. Принципы и основные цели применения SOA. Уровни SOA. Веб-сервисы. Платформы и средства создания веб-сервисов. Основные правила SOA. Протоколы HTTP, REST, SOAP, WSDL. Приводится описание стандарта WSDL. Рассматриваются средства BPEL и BPMN, Windows Workflow Foundation для создания сетей веб-сервисов.

Перечень лабораторных работ по разделу:

1. Реализация REST и SOAP веб-сервисов.
2. Создание сети веб-сервисов с применением технологии Windows Workflow Foundation.

*Структура дисциплины
по разделам и формам организации обучения*

Название раздела/темы	Аудиторная работа	СРС	Колл,	Итого
-----------------------	-------------------	-----	-------	-------

	(час)			(час)	Контр. Р.	
	Лек ции	Практ. /сем. зан.	Лаб. зан.			
1. Современные тенденции в области проектирования программных приложений. Архитектурные стили. Проектирование в жизненном цикле разработки программного обеспечения.	2		4	30		
2. Способы документирования архитектуры программного обеспечения. Software Design Document. Подходы к документированию Views and beyond и 4+1 design views.	2		6	30	Тест	
3. Проектирование модели предметной области. Подход Domain driven design.	2		10	38	Тест	
4. Архитектура веб-приложений.	2		10	40	Тест	
5. Сервис-ориентированная архитектура (SOA).	2		6	30	Тест	
Итого	16		32	168		216

6. ОРГАНИЗАЦИЯ И УЧЕБНО-МЕТОДИЧЕСКОЕ ОБЕСПЕЧЕНИЕ САМОСТОЯТЕЛЬНОЙ РАБОТЫ СТУДЕНТОВ

6.1 Виды и формы самостоятельной работы

Самостоятельная работа студентов включает текущую и творческую проблемно-ориентированную самостоятельную работу (ТСР).

Текущая СРС, направленная на углубление и закрепление знаний, а также развитие практических умений заключается в:

- работе студентов с лекционным материалом, поиске и анализе литературы и электронных источников информации по заданной проблеме и выбранной теме индивидуального задания,
- выполнении домашних заданий,
- изучении тем, вынесенных на самостоятельную проработку,
- изучении теоретического материала к лабораторным занятиям,
- подготовке к экзамену.

Творческая проблемно-ориентированная самостоятельная работа

(ТСР), направлена на развитие интеллектуальных умений, комплекса универсальных (общекультурных) и профессиональных компетенций, повышение творческого потенциала студентов и заключается в:

- поиске, анализе, структурировании и презентации информации, анализе научных публикаций по определенной теме исследований,
- презентация информации по теме реферата для студентов потока в конце лекций.

6.2 Контроль самостоятельной работы

Оценка результатов самостоятельной работы организуется как единство двух форм: самоконтроль и контроль со стороны преподавателя.

Самоконтроль в обучающей программе, контроль знаний, полученных с помощью обучающей программы.

Защита реферата, выступление с докладом.

По результатам текущего и рубежного контроля формируется допуск студента к экзамену. Экзамен проводится в письменной форме и оценивается преподавателем.

7. СРЕДСТВА ТЕКУЩЕЙ И ИТОГОВОЙ ОЦЕНКИ КАЧЕСТВА ОСВОЕНИЯ МОДУЛЯ

Оценка качества освоения дисциплины производится по результатам следующих контролирующих мероприятий:

Контролирующие мероприятия	Результаты обучения по дисциплине
Выполнение и защита отчетов по лабораторным работам	РД2 – РД3
Контроль выполнения индивидуального задания и его защита	РД3
Тестирование	РД2 – РД3
Проведение экзамена	РД1 – РД3

Для оценки качества освоения дисциплины при проведении контролирующих мероприятий предусмотрены следующие средства:

- вопросы для контролирующего теста;
- вопросы для экзамена.

7.1. Текущий контроль

Текущий контроль изучения дисциплины состоит из следующих видов:

- контроль за своевременным и правильным выполнением лабораторных работ и сдачей отчетов;
- контроль усвоения теоретического материала – проведение контрольных работ.

Контрольные проводятся в часы лекционных занятий, в письменной форме и включают задания по предыдущей лекции. Вопросы, вынесенные на контрольные работы, составлены лектором. По каждому варианту контрольной работы имеются 5-10 вопросов.

Примеры вопросов к контрольным:

1. Перечислите представления архитектуры программного продукта, которые создаются в рамках подхода «4+1 Design View». Укажите какие UML-диаграммы могут быть использованы для каждого из представлений
2. Для чего служит раздел bindings документа WSDL?
3. Каким типам http-запросов соответствуют операции CRUD при реализации REST веб-сервисов?
4. Приведите сценарии, при которых использование SOAP-сервисов более рационально, чем REST-сервисов.
5. При проектировании модели предметной области по каким признаком можно отличить сущность от ценного объекта?
6. Перечислите причины, приводящие к созданию аналитической модели предметной области, которая не может быть реализована в программном коде?
7. На каком архитектурном стиле основаны приложения, работающие по протоколу P2P (Torrent)?
 - а) Сервисно-ориентированный
 - б) Многослойный
 - в) Клиент-сервер
 - г) Многоуровневый
8. Что является идентификатором веб-сервиса?
 - а) Строка URL
 - б) IP-адрес сервера и номер порта
 - в) IP-адресом сервер, номер порта и название используемого сетевого протокола

7.2. Итоговый контроль

Итоговый контроль по дисциплине осуществляется по результатам выполнения лабораторных и индивидуальных работ и сдачи экзамена.

Примерный перечень экзаменационных вопросов:

1. Современные тенденции проектирования информационных систем. Принципы. Подходы. Архитектурные стили.
2. Средства представления проекта программного приложения и программной системы. Software Design Document. Views and beyond. 4+1 design views.

3. Микроархитектура и макроархитектура. Базовая и пилотная архитектуры.
4. Создание и реализация модели предметной области. Сущность, объект значения, сервис, репозиторий, агрегат.
5. Понятие сервисно-ориентированной архитектуры (SOA). Веб-сервисы.
6. Понятие сервисно-ориентированной архитектуры (SOA). Принципы и основные цели применения SOA. Уровни SOA.
7. Веб-сервисы. Платформы и средства создания веб-сервисов
8. Веб-сервисы. Протокол REST.
9. Веб-сервисы. Формат SOAP.
10. Веб-сервисы. WSDL.
11. Сети веб-сервисов. Технологии их реализации.

Пример экзаменационного билета:

1. (10 б.) Спроектируйте модульные представления is-part-of (модуль-подмодуль) и зависимостей между модулями для программного приложения Карта метеоусловий. Требование: не менее 5 модулей.
2. (10 б.) Приведен следующий программный код. Постройте диаграммы UML Sequence Diagram (диаграмма последовательностей) и UML Class Diagram (диаграмма классов)

```
class Team
{
    List<TeamMember> Members = new List<TeamMember>();
    public void NotifyOthers() { Console.WriteLine("Team.NotifyOthers"); ; }
    public void AddMember(TeamMember m)
    {
        Console.WriteLine("Team.AddMember");
        Members.Add(m);
        m.SetMyTeam(this);
    }
}

class TeamMember
{
    public void SetMyTeam(Team t)
    {
        Console.WriteLine("TeamMember.SetMyTeam");
        t.NotifyOthers();
    }
}

class App
{
    public static void Main()
    {
        Team t = new Team();
        TeamMember m1 = new TeamMember();
        t.AddMember(m1);
        Console.ReadKey();
    }
}
```

3. (5 б.) Сопоставьте CRUD-операции типам HTTP-запросов REST веб-сервиса.

4. (5 б.) Изобразите схему SOAP-конверта. Укажите назначение его составных частей.
5. (5 б.) Укажите, какие из архитектурных решений относятся к микроархитектуре:
 - a. Шаблон проектирования Синглентон
 - b. Сервис-ориентированная архитектура
 - c. Шаблон проектирования Фасад
6. (5 б.) Какое архитектурное решение позволяет снизить нагрузку на СУБД путем одновременного внесения изменений для группы записей?
 - a. Веб-сервис
 - b. Unit of work
 - c. Шаблон проектирования Синглетон

8. РЕЙТИНГ КАЧЕСТВА ОСВОЕНИЯ ДИСЦИПЛИНЫ

Оценка качества освоения дисциплины в ходе текущей и промежуточной аттестации обучающихся осуществляется в соответствии с «Руководящими материалами по текущему контролю успеваемости, промежуточной и итоговой аттестации студентов Томского политехнического университета», утвержденными приказом ректора № 88/од от 29.11.2013 г.

В соответствии с «Календарным планом изучения дисциплины»:

– текущая аттестация (оценка качества усвоения теоретического материала (ответы на вопросы и др.) и результаты практической деятельности (решение задач, выполнение заданий, решение проблем и др.) производится в течение семестра (оценивается в баллах (максимально 60 баллов), к моменту завершения семестра студент должен набрать не менее 33 баллов);

– промежуточная аттестация (экзамен, зачет) производится в конце семестра (оценивается в баллах (максимально 40 баллов), на экзамене (зачете) студент должен набрать не менее 22 баллов).

Итоговый рейтинг по дисциплине определяется суммированием баллов, полученных в ходе текущей и промежуточной аттестаций. Максимальный итоговый рейтинг соответствует 100 баллам.

В соответствии с «Календарным планом выполнения курсового проекта (работы)»:

– текущая аттестация (оценка качества выполнения разделов и др.) производится в течение семестра (оценивается в баллах (максимально 40 баллов), к моменту завершения семестра студент должен набрать не менее 22 баллов);

– промежуточная аттестация (защита проекта (работы)) производится в конце семестра (оценивается в баллах (максимально 60 баллов), по результатам защиты студент должен набрать не менее 33 баллов).

Итоговый рейтинг выполнения курсового проекта (работы) определяется суммированием баллов, полученных в ходе текущей и промежуточной аттестаций. Максимальный итоговый рейтинг соответствует 100 баллам (при наличии курсового проекта).

9. УЧЕБНО-МЕТОДИЧЕСКОЕ И ИНФОРМАЦИОННОЕ ОБЕСПЕЧЕНИЕ ДИСЦИПЛИНЫ

Основная литература:

1. Руководство Microsoft по проектированию архитектуры приложений. 2е издание. 2010. Дата обновления: 01.01.2010. URL: <http://apparchguide.ms/Book> (дата обращения: 01.09.2013).
2. Clements P., Bachmann F., Bass L. Documenting Software Architectures: Views and Beyond (2nd Edition). - Boston: Addison-Wesley Professional, 2010.
3. Kurtz, Jamie. ASP.NET MVC 4 and the Web API. – NY: Appress, 2013. – 141 p.
4. Betts D., Melnik G, Simonazzi F. Dependency Injection with Unity (Microsoft patterns & practices). - Redmond, USA: Microsoft Press, 2013.
5. Zimmermann O, Tomlinson M., Peuser S. Perspectives on Web Services: Applying SOAP, WSDL and UDDI to Real-World Projects. – NY:Springer, 2013. – 643 p.
6. Seemann M. Dependency Injection in .NET. – NY: Manning Publications, 2011. – 584 p.
7. Goma H. Software Modeling and Design: UML, Use Cases, Patterns, and Software Architectures. – Cambridge, UK: Cambridge University Press, 2011. – 578 p.
8. Kalin M. Java Web Services: Up and Running– NY: O'Reilly Media, 2009. – 360 p.
9. Bewis T. C# Design Pattern Essentials. – Southend-on-Sea: UK: Ability First Limited, 2012. – 264 p.
10. Arking J., Millett S. Professional Enterprise .NET. – NY: Wrox, 2009. – 504 p.

Дополнительная литература:

1. Freeman A. Pro ASP.NET MVC 5 (Expert's Voice in ASP.Net). – NY: Apress, 2013. – 832 p.

Программное обеспечение и Internet-ресурсы:

1. Платформа Java
2. BPM система Unify NXJ v12.0C (лицензионная версия для ВУЗов)
3. Oracle BPEL Process Manager
4. <http://fet.aics.ru/files/it-2009-04.zip>
5. <http://fet.aics.ru/metod/it/www/slides/toc.html>
6. <http://bpms.ru>
7. <http://www.intuit.ru/departement/internet/mwebtech/20/>
8. <http://www.service-architecture.com>
9. <http://www.w3.org/>
10. <http://www.oasis-open.org>
11. <http://www-128.ibm.com/developerworks/edu/x-dw-xml-i.html>

10. МАТЕРИАЛЬНО-ТЕХНИЧЕСКОЕ ОБЕСПЕЧЕНИЕ МОДУЛЯ

№ п/п	Наименование (компьютерные классы, учебные лаборатории, оборудование)	Корпус, ауд., количество установок
1	Компьютерный класс. 22 ПК Intel(R) Core(TM)2 CPU 6420, 2.13GHz, 2.14 ГГц, 2.00Гб ОЗУ	г. Томск, пр.Ленина, 2, учебный корпус №10, ауд.108, 109, 22 ПК.
2	платформы Java, .NET Framework, BPM система Unify NXJ v12.0C (лицензионная версия для ВУЗов).	

Программа составлена на основе Стандарта ООП ТПУ в соответствии с требованиями ФГОС ВО 3+ по направлению 09.04.01 «Информатика и вычислительная техника», магистерская подготовка.

Программа одобрена на заседании кафедры АиКС

(протокол No 10 от 30.05.2016)

Авторы

доцент кафедры АиКС

ассистент кафедры АиКС




Лунева Елена Евгеньевна

Баночкин Павел Иванович

Рецензент

доцент кафедры АиКС



Кочегурова Елена Алексеевна