

УТВЕРЖДАЮ

Проректор по учебной работе  
учреждения образования  
«Полоцкий государственный  
университет»



Ю.П. Голубев

20 19

Регистрационный № УД-250/19/уч.

## ТЕХНОЛОГИЯ РАЗРАБОТКИ ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ

Учебная программа учреждения высшего образования  
по учебной дисциплине для специальности  
1-98 01 01-01 «Компьютерная безопасность  
(математические методы и программные системы)»

Учебная программа составлена на основе образовательного стандарта по специальности высшего образования ОСВО 1-98 01 01-2013 и учебного плана по специальности 1-98 01 01-01 «Компьютерная безопасность (математические методы и программные системы)» (регистрационный №13-13/уч. ФИТ от 29.08.2013)



**СОСТАВИТЕЛЬ:**

Татьяна Сергеевна Рудькова, старший преподаватель кафедры технологий программирования учреждения образования «Полоцкий государственный университет»

**РЕКОМЕНДОВАНА К УТВЕРЖДЕНИЮ:**

Кафедрой технологий программирования учреждения образования «Полоцкий государственный университет»  
(протокол № 7 от 21.06. 2019)

Методической комиссией факультета информационных технологий учреждения образования «Полоцкий государственный университет»  
(протокол № 5 от 26.06. 2019)

## ПОЯСНИТЕЛЬНАЯ ЗАПИСКА

В настоящее время компьютеры все более широко используются во всех сферах деятельности человека. Функции, выполняемые современным программным обеспечением (ПО), становятся все более сложными и разнообразными. С учетом этого растут размеры и сложность разрабатываемого ПО, возрастают требования к их качеству.

Создание такого ПО без знания и использования основ организации процесса их разработки, а также современных методологий, технологий и инструментальных средств разработки, является невозможным. Именно на такой подход к разработке ПО направлено изучение дисциплины «Технологии разработки программного обеспечения», поэтому данная учебная дисциплина является актуальной.

*Целью* преподавания дисциплины является обучение современным методологиям и технологиям разработки ПО, а также получение студентами знаний и умений в проектировании и построении процесса разработки ПО.

Достижение поставленной цели предполагает решение следующих *задач*:

- формирование базовых знаний в области стратегий разработки ПО, моделей жизненного цикла (ЖЦ), реализующих данные стратегии, и выбора моделей ЖЦ для конкретного проекта;
- формирование базовых знаний в области основных методологий разработки ПО;
- формирование базовых знаний в области CASE-технологий разработки ПО.

В результате изучения дисциплины «Технология разработки программного обеспечения» формируются следующие *компетенции*:

академические компетенции:

АК-1. Уметь применять базовые научно-теоретические знания для решения теоретических и практических задач.

АК-3. Владеть исследовательскими навыками.

АК-4. Уметь работать самостоятельно.

АК-5. Быть способным выработать новые идеи (креативность).

АК-6. Владеть междисциплинарным подходом при решении проблем.

АК-7. Иметь навыки, связанные с использованием технических устройств, управлением информацией и работой с компьютером.

АК-9. Уметь учиться, повышать свою квалификацию в течение всей жизни.

социально-личностные компетенции:

СЛК-2. Быть способным к социальному взаимодействию.

СЛК-3. Обладать способностью к межличностным коммуникациям.

СЛК-5. Быть способным к критике и самокритике (критическое мышление).

профессиональные компетенции:

ПК-8. Взаимодействовать со специалистами смежных профилей.

ПК-12. Пользоваться глобальными информационными ресурсами.

ПК-24. Работать с научной, технической и патентной литературой.

В результате изучения дисциплины студент должен

*знать:*

- стратегии и модели жизненного цикла ПО;
- классические и гибкие методологии разработки ПО;
- основы проектирования ПО;
- основы организации процесса разработки ПО;
- современные методы и технологии разработки ПО;
- основные архитектуры современных средств разработки ПО;

*уметь:*

- выбрать модель жизненного цикла ПО, исходя из особенностей конкретного проекта;
- проектировать структуру и архитектуру разрабатываемого ПО;
- разрабатывать диаграммы UML;

*владеть:*

- основами CASE-систем;
- навыками разработки программных компонент с использованием технологий и визуальных сред программирования.

Базовыми учебными дисциплинами по курсу «Технология разработки программного обеспечения» являются «Программирование», «Стандартизация программного обеспечения» и «Технологии разработки и защиты серверных веб-приложений и веб-служб». В свою очередь учебная дисциплина «Технология разработки программного обеспечения» является базой для выполнения дипломных работ по данному направлению.

Согласно учебному плану учебная программа изучения дисциплины «Технология разработки программного обеспечения» рассчитана следующим образом:

<b>Форма получения высшего образования</b>	<b>дневная</b>
Курс	4
Семестр	7
Всего часов по учебной дисциплине	104
Аудиторных часов по учебной дисциплине	36
В том числе:	
Лекции, часов	18
Практические занятия, часов	18
Самостоятельная работа студентов	68
Трудоемкость дисциплины, зачетные единицы	2,5
Форма текущей аттестации	экзамен
Курсовой проект, часов	40
Трудоемкость курсового проектирования, зачетные единицы	1

## СОДЕРЖАНИЕ УЧЕБНОГО МАТЕРИАЛА

### РАЗДЕЛ 1. ЖИЗНЕННЫЙ ЦИКЛ ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ И ЕГО МОДЕЛИ

Тема 1. Стандартизация жизненного цикла, стратегии разработки и модели жизненного цикла программного обеспечения

Понятие программного обеспечения. Классификация программного обеспечения. Понятие о технологии разработки программного обеспечения. Понятие жизненного цикла программного обеспечения. Стандартизация жизненного цикла программного обеспечения. Процессы жизненного цикла программного обеспечения: их виды и взаимосвязь. Сертификация и оценка качества процессов разработки программного обеспечения: подходы к оценке, применение.

Понятие стратегии разработки программного обеспечения. Виды стратегий разработки и их сравнительная характеристика. Модели жизненного цикла: каскадная модель, V-образная модель, макетирование (прототипирование), инкрементная модель, спиральная модель, модель быстрой разработки программного обеспечения. Организация моделей жизненного цикла, их преимущества, недостатки и критерии использования. Адаптированные модели жизненного цикла и их использование. Выбор модели жизненного цикла для выполнения конкретного программного проекта.

### РАЗДЕЛ 2. ОСНОВЫ АНАЛИЗА И ПРОЕКТИРОВАНИЯ ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ

Тема 2. Основы структурного и объектно-ориентированного анализа и проектирования, визуальное моделирование

Анализ и проектирование: назначение этапов, их цели и содержание. Структурный анализ и проектирование: основные понятия и принципы. Объектно-ориентированный анализ проектирование: основные понятия и принципы.

Понятие моделирование. Особенности моделирования информационных систем и программного обеспечения. Понятие визуального моделирования. Средства визуального моделирования. Языки визуального моделирования: понятие, состав языка.

### РАЗДЕЛ 3. УНИФИЦИРОВАННЫЙ ЯЗЫК МОДЕЛИРОВАНИЯ

Тема 3.1. История происхождения и развития языка UML

Причины, приведшие к появлению языка унифицированного моделирования. Методологии, положенные в основу языка UML. История развития языка UML. Основные элементы языка: иерархия уровней, пакеты ядра, диаграммы и их виды. Механизмы расширений языка UML.

Тема 3.2. Диаграмма вариантов использования

Назначение диаграммы. Предметы диаграммы и их графическое обозначение: вариант использования, граница, примечание. Отношения диаграмм, обозначение: ассоциация, включение, расширение, обобщение.

Принципы и порядок построения диаграммы вариантов использования. Понятие сценария и его описание.

Определение весовых коэффициентов сложности вариантов использования и актеров. Расчет нескорректированных показателей Use Case. Определение технической сложности проекта. Определение уровня квалификации разработчиков. Расчет скорректированных показателей Use Case. Определение продолжительности и затрат на разработку проекта.

Тема 3.3. Диаграммы классов, композитной структуры, пакетов и объектов

Назначение диаграммы. Понятие класса и его графическое обозначение. Синтаксис описания атрибутов и операций класса. Отношение ассоциации между классами: бинарная ассоциация, исключая ассоциация, N-арная ассоциация, ассоциация-класс. Отношение зависимости. Агрегация и композиция как подвиды ассоциации. Отношение обобщения и ограничения обобщений. Отношение реализации и его использование.

Назначение диаграммы композитной структуры. Предметы диаграммы: композитный класс, часть, порты, кооперации. Взаимосвязь частей композитного класса. Взаимодействие композитных классов через порты. Назначение диаграммы пакетов. Понятие пакета UML. Виды отношений между пакетами: включение, зависимость, импорт и слияние. Назначение диаграммы объектов. Понятие объекта. Отношения между объектами.

Тема 3.4. Диаграммы последовательностей и деятельности

Назначение диаграммы последовательностей. Понятия событий и сообщений, траектории событий. Линия жизни: понятие, графическое обозначение. Прием и передача сообщений между линиями жизни. Виды и типы сообщений, формат описания сообщения. Комбинированные фрагменты взаимодействия: понятие, назначение и графическое обозначение. Виды операторов комбинированных фрагментов. Задание временных ограничений на диаграммах последовательностей.

Назначение диаграммы деятельности. Действие и деятельность: понятие, графическое обозначение, отличие. Узлы объектов. Маркеры и их свойства. Дуги диаграммы деятельности и их виды. Узлы управления диаграммы деятельности и их применение. События приема сигнала и передача сигнала. Специальные виды объектов: центральный буфер и хранилище. Разбиения и их использование. Регионы прерываемой деятельности и обработчики прерываний.

Тема 3.5. Диаграммы конечных автоматов и дополнительные диаграммы взаимодействий

Назначение диаграммы конечных автоматов. Понятие конечного автомата и его свойства. Простое состояние: понятие и графическое обозначение. Простой переход: понятие и графическое обозначение. Псевдосостояния: понятие, виды и использование. Составной переход: понятие, правила срабатывания. Композитное состояние: понятие и графическое обозначение. Ортогональное композитное состояние. Правила входа и выхода для композитного состояния. Протокольный конечный автомат: понятие, правила построения и ограничения.

Назначение диаграммы коммуникации. Линия жизни: понятие и графическое обозначение. Взаимосвязь между линиями жизни, передача сообщений между линиями жизни. Назначение диаграммы обзора взаимодействий. Основные элементы и правила построения. Отличие от диаграмм деятельности и последовательности. Назначение временной диаграммы. Обозначение линий жизни на временной диаграмме. Особенности моделирования времени при передаче сообщений между линиями жизни.

### Тема 3.6. Диаграммы компонентов и развертывания

Назначение диаграммы компонентов. Понятие компонента, его графическое обозначение и виды. Понятие порта и его графическое обозначение. Понятие интерфейса, его графическое обозначение и виды. Делегирующий и собирающий соединители, их обозначения и правила использования. Зависимость между компонентами. Стандартные стереотипы компонентов.

Назначение диаграммы развертывания. Понятие узла развертывания, его графическое обозначение и виды. Понятие артефакта и его графическое обозначение. Отношения развертывания, манифестации и пути коммуникации.

## РАЗДЕЛ 4. УНИФИЦИРОВАННЫЙ ПРОЦЕСС И ГИБКИЕ ПРОЦЕССЫ РАЗРАБОТКИ ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ

Тема 4. Основы и характеристика процессов RUP и гибких процессов разработки программного обеспечения

Основные понятия унифицированного процесса разработки программного обеспечения. Особенность инкрементно-ориентированного подхода. Измерения: процессы и фазы. Краткая характеристика фаз унифицированного подхода. Краткая характеристика процессов RUP: управление проектом, бизнес-моделирование, управление требованиями, анализ и проектирование, реализация, тестирование, развертывание.

Понятие гибких процессов разработки программного обеспечения. Причины, приведшие к развитию гибких процессов. Обзор направлений развития гибких методологий. Принципы гибкой методологии Agile. Представление об экстремальном программировании. Представление о SCRUM-процессе. Функционально ориентированная разработка программного обеспечения.

## КУРСОВОЕ ПРОЕКТИРОВАНИЕ

Курсовой проект по данной учебной дисциплине – это самостоятельная программная разработка студента по заданной теме в области технологий разработки программного обеспечения.

Целью курсового проектирования является приобретение навыков самостоятельного творческого решения общетеоретических и практических вопросов разработки программ различного целевого назначения, начиная от технического задания на проект, анализа предметной области и заканчивая оформлением пояснительной записки и диаграмм.

Курсовой проект состоит из:

- пояснительной записки;
- графической части;
- работающего программного средства.

В пояснительной записке отражаются основные этапы разработки программного средства в следующих разделах:

Введение.

1. Анализ задания и постановка задач.
2. Создание функциональной модели ПО.
3. Создание логической модели ПО.
4. Создание физической модели ПО.
5. Описание реализации и тестирования ПО.

Заключение.

Литература.

В графической части отражаются схемы и чертежи, дополняющие и иллюстрирующие содержимое пояснительной записки.

Графическая часть состоит из:

- диаграмма вариантов использования;
- диаграмма классов;
- диаграммы последовательностей;
- диаграммы деятельности;
- диаграммы состояний;
- диаграмма компонентов;
- диаграмма развертывания.

Дополнительно в графическую часть могут быть внесены прочие диаграммы языка UML, схемы базы данных и экранные формы приложения.

### ПРИМЕРНЫЙ ПЕРЕЧЕНЬ ТЕМ КУРСОВЫХ ПРОЕКТОВ

1. Разработка информационной системы «Выставка-продажа картин».
2. Разработка информационной системы «Морское пароходство».
3. Разработка информационной системы автовокзала города.
4. Разработка информационной системы автомобильного хозяйства.



5. Разработка информационной системы автошколы.
6. Разработка информационной системы аптечной сети города.
7. Разработка информационной системы библиотек города.
8. Разработка информационной системы бронирования билетов в авиакомпании.
9. Разработка информационной системы городского агентства недвижимости.
10. Разработка информационной системы городского бюро по найму.
11. Разработка информационной системы деканат (студенты факультета).
12. Разработка информационной системы для работы приемной комиссии.
13. Разработка информационной системы кафедра (преподаватели и предметы).
14. Разработка информационной системы кинотеатров города.
15. Разработка информационной системы отдела кадров предприятия.
16. Разработка информационной системы питомника.
17. Разработка информационной системы проведения тестирования знаний (экзамены, зачеты).
18. Разработка информационной системы продажи товаров через Интернет-магазин.
19. Разработка информационной системы расписания занятий студентов.
20. Разработка информационной системы результатов экзаменационных сессий студентов.
21. Разработка информационной системы ресторана-бара.
22. Разработка информационной системы салона красоты.
23. Разработка информационной системы туристического агентства.
24. Разработка информационной системы составления рейтингов музыкальных исполнителей на музыкальном телеканале.
25. Разработка информационной системы станции техобслуживания.
26. Разработка информационной системы телефонной сети города.
27. Разработка информационной системы учета материальных ценностей на складе.
28. Разработка информационной системы учета проживания в гостинице.
29. Разработка информационной системы учета проживания студентов в общежитии.
30. Разработка информационной системы учета сдачи контрольных и курсовых работ студентами заочной формы обучения.

**Учебно-методическая карта дисциплины для студентов дневной формы получения образования**

Номер раздела, темы	Название раздела, темы	Количество аудиторных часов				Литература	Формы контроля знаний
		лекции	практические занятия	лабораторные занятия	управляемая самостоятельная работа студента		
1	2	3	4	5	6	7	8
<b>РАЗДЕЛ 1. ЖИЗНЕННЫЙ ЦИКЛ ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ И ЕГО МОДЕЛИ</b>							
Тема 1.	Стандартизация жизненного цикла, стратегии разработки и модели жизненного цикла программного обеспечения	2				[1,2,3,4,5]	УО <sup>1</sup>
<b>РАЗДЕЛ 2. ОСНОВЫ АНАЛИЗА И ПРОЕКТИРОВАНИЯ ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ</b>							
Тема 2.	Основы структурного и объектно-ориентированного анализа и проектирования, визуальное моделирование	2				[1,2,3,5,6]	УО
<b>РАЗДЕЛ 3. УНИФИЦИРОВАННЫЙ ЯЗЫК МОДЕЛИРОВАНИЯ</b>							
Тема 3.1.	История происхождения и развития языка UML	2				[3, 6,8,10]	УО
Тема 3.2.	Диаграмма вариантов использования	2				[3,4,6,7,8]	УО
	Практическое занятие № 1. Разработка диаграмм вариантов использования		4			МУ <sub>пу</sub> <sup>2</sup>	ПУ <sup>3</sup>
	Практическое занятие № 2. Проектирование графического пользовательского интерфейса		2			МУ <sub>пу</sub>	ПУ

<sup>1</sup>УО – устный опрос на занятии

<sup>2</sup>МУ<sub>пу</sub> – методические указания по выполнению практических упражнений

<sup>3</sup>ПУ – письменный отчет по практическим упражнениям с их устной защитой

1	2	3	4	5	6	7	8
Тема 3.3.	Диаграммы классов, композитной структуры, пакетов и объектов	2				[3,4,6,7,8]	УО
	Практическое занятие № 3. Разработка диаграмм классов		2			МУ <sub>пу</sub>	ПУ
Тема 3.4.	Диаграммы последовательностей и деятельности	2				[3,4,6,7,8]	УО
	Практическое занятие № 4. Разработка диаграмм последовательностей		2			МУ <sub>пу</sub>	ПУ
	Практическое занятие № 5. Разработка диаграмм деятельности		2			МУ <sub>пу</sub>	ПУ
Тема 3.5.	Диаграммы конечных автоматов и дополнительные диаграммы взаимодействий	2				[4,5,6,7,8]	УО
	Практическое занятие № 6. Разработка диаграмм состояний		2			МУ <sub>пу</sub>	ПУ
Тема 3.6.	Диаграммы компонентов и развертывания	2				[4,5,6,7,8]	УО
	Практическое занятие № 7. Разработка диаграмм компонентов		2			МУ <sub>пу</sub>	ПУ
	Практическое занятие № 8. Разработка диаграмм развертывания		2			МУ <sub>пу</sub>	ПУ
<b>РАЗДЕЛ 4. УНИФИЦИРОВАННЫЙ ПРОЦЕСС И ГИБКИЕ ПРОЦЕССЫ РАЗРАБОТКИ ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ</b>							
Тема 4.	Основы и характеристика процессов RUP и гибких процессов разработки программного обеспечения	2				[3,4,7,9]	УО
<b>Итого:</b>		<b>18</b>	<b>18</b>				

## ИНФОРМАЦИОННО-МЕТОДИЧЕСКАЯ ЧАСТЬ

## Литература

## Основная:

1. Введение в программные системы и их разработку [Электронный ресурс] / С.В. Назаров, С.Н. Белоусова, И.А. Бессонова и др. – 2-е изд., испр. – Москва : Национальный Открытый Университет «ИНТУИТ», 2016. – 650 с. – Режим доступа: <http://biblioclub.ru/index.php?page=book&id=429819>.
2. Рак, И.П. Основы разработки информационных систем [Электронный ресурс] : учебное пособие / И.П. Рак, А.В. Платёнкин, А.В. Терехов ; Министерство образования и науки Российской Федерации, Федеральное государственное бюджетное образовательное учреждение высшего профессионального образования «Тамбовский государственный технический университет». – Тамбов : ФГБОУ ВПО «ТГТУ», 2017. – 99 с. – Режим доступа: <http://biblioclub.ru/index.php?page=book&id=499041>.
3. Долженко, А.И. Технологии командной разработки программного обеспечения информационных систем [Электронный ресурс] / А.И. Долженко. – 2-е изд., исправ. – Москва : Национальный Открытый Университет «ИНТУИТ», 2016. – 301 с. – Режим доступа: <http://biblioclub.ru/index.php?page=book&id=428801>.
4. Кулямин, В. Компонентный подход в программировании / В. Кулямин. – 2-е изд., исправ. – Москва : Национальный Открытый Университет «ИНТУИТ», 2016. – 591 с. – Режим доступа: <http://biblioclub.ru/index.php?page=book&id=429086>.
5. Ипатова, Э.Р. Методологии и технологии системного проектирования информационных систем [Электронный ресурс] : учебник / Э.Р. Ипатова, Ю.В. Ипатов. – 2-е изд., стер. – Москва : Флинта, 2016. – 257 с. – Режим доступа: <http://biblioclub.ru/index.php?page=book&id=79551>.
6. Влацкая, И.В. Проектирование и реализация прикладного программного обеспечения [Электронный ресурс] : учебное пособие / И.В. Влацкая, Н.А. Заельская, Н.С. Надточий ; Министерство образования и науки Российской Федерации, Федеральное государственное бюджетное образовательное учреждение высшего профессионального образования «Оренбургский государственный университет», Кафедра компьютерной безопасности и математического обеспечения информационных систем. – Оренбург : ОГУ, 2015. – 119 с. – Режим доступа: <http://biblioclub.ru/index.php?page=book&id=439107>.
7. Соловьев, Н. Системы автоматизации разработки программного обеспечения [Электронный ресурс] : учебное пособие / Н. Соловьев, Е. Чернопрудова ; Министерство образования и науки Российской Федерации, Федеральное государственное бюджетное образовательное учреждение высшего профессионального образования «Оренбургский государственный университет». – Оренбург : ОГУ, 2012. – 191 с. – Режим доступа: <http://biblioclub.ru/index.php?page=book&id=270302>.

8. Леоненков, А. Нотация и семантика языка UML [Электронный ресурс] / А. Леоненков. – 2-е изд., исправ. – Москва : Национальный Открытый Университет «ИНТУИТ», 2016. – 205 с. – Режим доступа: <http://biblioclub.ru/index.php?page=book&id=429143>.

9. Гибкая методология разработки программного обеспечения [Электронный ресурс] : курс / Национальный Открытый Университет «ИНТУИТ». – Москва : Интернет-Университет Информационных Технологий, 2010. – 134 с. – Режим доступа: <http://biblioclub.ru/index.php?page=book&id=233769>.

Дополнительная:

10. Хританков, А.С. Проектирование на UML [Электронный ресурс] : сборник задач / А.С. Хританков, В.А. Полежаев, А.И. Андрианов. – 3-е изд. стер. – Москва ; Берлин : Директ-Медиа, 2018. – 242 с. – Режим доступа: <http://biblioclub.ru/index.php?page=book&id=483549>.

11. Волкова, Т.В. Проектирование компонентов автоматизированных систем в примерах [Электронный ресурс] : учебное пособие / Т.В. Волкова, Е.Н. Чернопрудова ; Оренбургский Государственный Университет, Кафедра программного обеспечения вычислительной техники и автоматизированных систем. – Оренбург : Оренбургский государственный университет, 2017. – 178 с. – Режим доступа: <http://biblioclub.ru/index.php?page=book&id=481817>.

12. Проектирование информационных систем. Проектный практикум [Электронный ресурс] : учебное пособие / А.В. Платёнкин, И.П. Рак, А.В. Терехов, В.Н. Чернышов ; Министерство образования и науки Российской Федерации, Федеральное государственное бюджетное образовательное учреждение высшего профессионального образования «Тамбовский государственный технический университет». – Тамбов : Издательство ФГБОУ ВПО «ТГТУ», 2015. – 81 с. – Режим доступа: <http://biblioclub.ru/index.php?page=book&id=444966>.

13. Малышева, Е.Н. Проектирование информационных систем (Раздел 5. Индустриальное проектирование информационных систем. Объектно-ориентированная Case-технология проектирования информационных систем) [Электронный ресурс]: учебное пособие / Е.Н. Малышева. – Кемерово : Кемеровский государственный университет культуры и искусств (КемГУКИ), 2009. – 70 с. – Режим доступа: <http://biblioclub.ru/index.php?page=book&id=227740>.

14. Бабич, А.В. UML. Первое знакомство [Электронный ресурс]: Пособие для подготовки к сдаче теста UM0-100 / А.В. Бабич ; Национальный Открытый Университет «ИНТУИТ». – Москва : Интернет-Университет Информационных Технологий, 2008. – 176 с. – Режим доступа: <http://biblioclub.ru/index.php?page=book&id=233305>.

15. Котляров, В.П. Основы тестирования программного обеспечения [Электронный ресурс] / В.П. Котляров, Т.В. Коликова. – Москва : Интернет-Университет Информационных Технологий, 2009. – 288 с. – Режим доступа: <http://biblioclub.ru/index.php?page=book&id=233107>.

## ПРИМЕРНЫЙ ПЕРЕЧЕНЬ ПРАКТИЧЕСКИХ ЗАНЯТИЙ

1. Разработка диаграмм вариантов использования.
2. Проектирование графического пользовательского интерфейса.
3. Разработка диаграмм классов.
4. Разработка диаграмм последовательностей.
5. Разработка диаграмм деятельностей.
6. Разработка диаграмм состояний.
7. Разработка диаграмм компонентов.
8. Разработка диаграмм развертывания.

## ПЕРЕЧЕНЬ КОМПЬЮТЕРНЫХ ПРОГРАММ, НАГЛЯДНЫХ И ДРУГИХ ПОСОБИЙ, МЕТОДИЧЕСКИХ УКАЗАНИЙ И МАТЕРИАЛОВ, ТЕХНИЧЕСКИХ СРЕДСТВ ОБУЧЕНИЯ, ОБОРУДОВАНИЯ ДЛЯ ВЫПОЛНЕНИЯ ПРАКТИЧЕСКИХ УПРАЖНЕНИЙ

1. ПЭВМ с основной конфигурацией AMD Athlon II 64 X4 / DDR3 4GB/ 500GB/.
2. Операционная система – Windows 7+.
3. Case-средства – All Fusion Process Modeler 7+, ErWin Data Modeler 7+, Enterprise Architect 8+.

## ПРИМЕРНЫЙ ПЕРЕЧЕНЬ ВОПРОСОВ ДЛЯ ПРОВЕДЕНИЯ ЭКЗАМЕНА

1. Понятие программного обеспечения, классификация программного обеспечения.
2. Жизненный цикл ПО и его стандартизация, процессы ЖЦ ПО, группы процессов ЖЦ ПО.
3. Процесс разработки ПО: основные действия и их содержание.
4. Сертификация процессов разработки ПО, модель СММ.
5. Стратегии жизненного цикла ПО: понятие, виды и их сравнительная характеристика.
6. Каскадная модель жизненного цикла ПО: описание, преимущества и недостатки, критерии применения.
7. V-образная модель жизненного цикла ПО: описание, преимущества и недостатки, критерии применения.
8. Инкрементная модель жизненного цикла ПО: описание, преимущества и недостатки, критерии применения.
9. Спиральная модель жизненного цикла ПО: описание, преимущества и недостатки, критерии применения.
10. RAD модель жизненного цикла ПО: описание, преимущества и недостатки, критерии применения.
11. Основные этапы проектирования программных систем и их содержание.

12. Объектно-ориентированный подход к разработке ПО: основные понятия и принципы.
13. Язык UML: причины появления и история развития языка, структура языка.
14. Канонические диаграммы языка UML: их виды и типы, рекомендации построения.
15. Диаграмма вариантов использования: назначение, принципы построения.
16. Оценка трудоемкости разработки проекта на основе вариантов использования.
17. Диаграмма классов: назначение, классы, обозначение классов, их атрибутов и операций.
18. Диаграмма классов: назначение, отношения между классами и их применение.
19. Диаграмма композитной структуры: композитные классы и их части, принципы построения, кооперации и их использование.
20. Диаграмма пакетов: назначение, пакеты и отношения между ними.
21. Диаграмма последовательности: назначение, линии жизни.
22. Диаграмма деятельности: назначение, понятие, семантика и обозначение деятельности, действия и дуг.
23. Диаграмма деятельности: узлы управления, их виды и применение.
24. Дополнительные элементы диаграммы деятельности: действия приема и передачи сигналов, центральный буфер и хранилище данных.
25. Диаграмма конечного автомата: назначение, простое и композитное состояния.
26. Диаграмма конечного автомата: простые и составные переходы, правила срабатывания переходов.
27. Диаграмма конечного автомата: псевдосостояния, их виды и применение.
28. Диаграмма компонентов: назначение, компоненты, интерфейсы и порты, соединения и их виды.
29. Диаграмма развертывания: назначение, узлы, артефакты, соединения и их виды.
30. Проблемы классического подхода к разработке ПО и причины появления гибких методологий.
31. Краткая характеристика процессов RUP.
32. Гибкие методологии. Преимущества и область применения.
33. Экстремальное программирование: понятие, базис XP, структура XP цикла разработки.
34. Принципы гибкой методологии Agile.
35. SCRUM процесс: понятие, роли, мероприятия, уровни команд в SCRUM.
36. Функционально ориентированная разработка программного обеспечения.

37. CASE-средства: понятие, история появления и развития, структура и состав, классификация.

38. CASE-средства: структура и состав, классификация.

### ОРГАНИЗАЦИЯ САМОСТОЯТЕЛЬНОЙ РАБОТЫ СТУДЕНТОВ

При изучении дисциплины студентами дневной формы получения образования используются следующие формы самостоятельной работы:

- подготовка к устным опросам на лекции;
- подготовка к защите отчетов по практическим упражнениям;
- систематизация полученных знаний при подготовке к экзамену.

#### Содержание самостоятельной работы студентов дневной формы получения образования

Вид самостоятельной работы	Тематическое содержание и используемые источники	Количество часов
1	2	3
Подготовка к устным опросам на лекции	Тема 1. Литература: [1,2,2,3,5]	1
	Тема 2. Литература:[1,2,3,5,6]	1
	Тема 3.1. Литература:[3,6,8,10]	2
	Тема 3.2. Литература:[3,4,6,7,8]	2
	Тема 3.3. Литература:[3,4,6,7,8]	2
	Тема 3.4. Литература:[3,4,6,7,8]	2
	Тема 3.5. Литература:[4,5,6,7,8]	2
	Тема 3.6. Литература:[4,5,6,7,8]	2
	Тема 4. Литература:[3,4,7,9]	2
Подготовка к защите отчетов по практическим упражнениям	Практическое занятие № 1 [МУ <sub>пу</sub> ]	2
	Практическое занятие № 2 [МУ <sub>пу</sub> ]	2
	Практическое занятие № 3 [МУ <sub>пу</sub> ]	2
	Практическое занятие № 4 [МУ <sub>пу</sub> ]	2
	Практическое занятие № 5 [МУ <sub>пу</sub> ]	2
	Практическое занятие № 6 [МУ <sub>пу</sub> ]	2
	Практическое занятие № 7 [МУ <sub>пу</sub> ]	2
	Практическое занятие № 8 [МУ <sub>пу</sub> ]	2
Систематизация полученных знаний при подготовке к экзамену		36
<b>Итого:</b>		<b>68</b>

Перечень дополнительного информационного и учебно-методического обеспечения самостоятельной работы студентов, размещенного в GoogleClassRoom университета:

1. Конспект лекций.
2. Методические указания по выполнению практических упражнений.
3. Методические указания к выполнению курсового проекта.



## СРЕДСТВА ДИАГНОСТИКИ РЕЗУЛЬТАТОВ УЧЕБНОЙ ДЕЯТЕЛЬНОСТИ

Диагностика результатов учебной деятельности осуществляется следующими средствами:

- устный опрос на лекции;
- отчет по практическим упражнениям с их устной защитой;
- курсовой проект;
- экзамен.

### КОНТРОЛЬ КАЧЕСТВА УСВОЕНИЯ ЗНАНИЙ

Контроль качества усвоения знаний проводится в соответствии с Положением о рейтинговой системе оценки знаний и компетенций студентов (приказ ректора университета от 06.06.2014 № 294 (в редакции, утвержденной приказом ректора университета от 17.11.2014 № 605) в форме промежуточного контроля и текущей аттестации.

Результат промежуточного контроля за семестр оценивается отметкой в баллах по десятибалльной шкале и выводится, исходя из отметок, выставленных в ходе проведения мероприятий промежуточного контроля в течение семестра по следующей формуле:

$$\Pi = \frac{(УО_1 + \dots + УО_9) + (ПУ_1 + \dots + ПУ_8)}{17}$$

где  $УО_1 + \dots + УО_9$  – отметки, выставленные по результатам устных опросов на лекциях;

$ПУ_1 + \dots + ПУ_8$  – отметки, выставленные по результатам устных защит отчетов по практическим упражнениям.

Результат промежуточного контроля рассчитывается как округленное среднее значение.

Текущая аттестация проводится в форме экзамена по представленным в программе вопросам.

Итоговая экзаменационная отметка по дисциплине рассчитывается по формуле:

$$ИЭ = k \cdot \Pi + (1 - k) \cdot О$$

где  $k$  – весовой коэффициент промежуточного контроля;

$\Pi$  – результат промежуточного контроля за семестр;

$О$  – отметка, полученная студентом на экзамене за ответ по билету.

Весовой коэффициент  $k$  принимается равным 0,5. Информация о весовом коэффициенте доводится до студентов на первом занятии в семестре. Положительной является экзаменационная отметка не ниже 4 баллов.

## ХАРАКТЕРИСТИКА ИННОВАЦИОННЫХ ПОДХОДОВ К ПРЕПОДАВАНИЮ УЧЕБНОЙ ДИСЦИПЛИНЫ

Изучение дисциплины осуществляется на лекционных и практических занятиях. На лекционных занятиях студенты овладевают системой теоретических знаний в технологии разработки программного обеспечения. В ходе лекционного изложения теоретических сведений используются: проблемно-модульное изложение материала; традиционные словесные приемы и методы, которые активизируются постановкой проблемных вопросов и заданий, организацией учебных дискуссий в опоре на имеющуюся начальную подготовку студентов и их политехнический кругозор; интерактивные методы обучения.

На практических занятиях развиваются и формируются необходимые практические умения и навыки по применению различного инструментария при разработке программного обеспечения.

ПРОТОКОЛ СОГЛАСОВАНИЯ УЧЕБНОЙ ПРОГРАММЫ  
С ДРУГИМИ УЧЕБНЫМИ ДИСЦИПЛИНАМИ СПЕЦИАЛЬНОСТИ

Название учебной дисциплины, с которой требуется согласование	Название кафедры	Предложения об изменениях в содержании учебной программы	Решение, принятое кафедрой, разработавшей учебную программу (с указанием даты и номера протокола)
Дипломная работа	Кафедра технологий программирования	<i>Нет</i>	

Заведующий кафедрой  
технологий программирования



О.Н. Петрович