

Документ подписан простой электронной подписью
Информация о владельце:
ФИО: Выборнова Любовь Алексеевна
Должность: Ректор
Дата подписания: 03.02.2022 15:17:47
Уникальный программный ключ:
с3b3b9c625f8e4a7b0e9a0a0c87d6

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ
ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ
ВЫСШЕГО ОБРАЗОВАНИЯ
«ПОВОЛЖСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ СЕРВИСА» (ФГБОУ ВО «ПВГУС»)

Кафедра «Информационный и электронный сервис»

РАБОЧАЯ УЧЕБНАЯ ПРОГРАММА

по дисциплине «Введение в программную инженерию»
для студентов направления подготовки 09.03.04 «Программная инженерия»
направленности (профиля) "Разработка программно-информационных систем"

Тольятти, 2018

Рабочая учебная программа по дисциплине «Введение в программную инженерию» включена в основную профессиональную образовательную программу направления подготовки 09.03.04 «Программная инженерия» направленности (профиля) "Разработка программно-информационных систем" решением Президиума Ученого совета

Протокол № 4 от 28.06.2018 г.

Начальник учебно-методического отдела _____  Н.М.Шемендюк
28.06.2018 г.

Рабочая учебная программа по дисциплине «Введение в программную инженерию» разработана в соответствии с Федеральным государственным образовательным стандартом направления подготовки 09.03.04 «Программная инженерия» утвержденный приказом Министерства образования и науки Российской Федерации от 12 марта 2015 г. N 229

Составил: д.т.н., профессор В.И. Воловач, к.т.н. Н.Г.Пудовкина

СОГЛАСОВАНО:

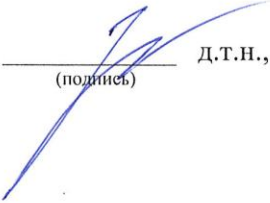
Директор научной библиотеки  В.Н.Еремина

СОГЛАСОВАНО:

Начальник управления информатизации  В.В.Обухов

Рабочая программа утверждена на заседании кафедры «Информационный и электронный сервис»

Протокол № 11 от «27» июня 2018 г.

Заведующий кафедрой 
(подпись) д.т.н., профессор В.И. Воловач

СОГЛАСОВАНО:

Начальник учебно-методического отдела  Н.М.Шемендюк

1. Перечень планируемых результатов обучения по дисциплине, соотнесенных с планируемыми результатами освоения образовательной программы

1.1. Цели освоения дисциплины

Целями освоения дисциплины являются

- получение студентами направления подготовки 09.03.04 «Программная инженерия» систематизированных представлений о современном комплексе задач, методов и стандартах программной инженерии, создании и эволюции сложных, многоверсионных, тиражируемых программных продуктах высокого качества;

- формирование у бакалавров целостной системы знаний о процессах, ведущих к созданию программного обеспечения: от начальной разработки системных требований и далее через проектирование, непосредственное программирование и аттестацию программных систем.

1.2. В соответствии с видами профессиональной деятельности, на которые ориентирована образовательная программа направления подготовки 09.03.04 «Программная инженерия», содержание дисциплины позволит обучающимся решать следующие профессиональные задачи:

организационно-управленческая деятельность:

- планирование и организация собственной работы;
- планирование и координация работ по настройке и сопровождению программного продукта;
- организация работы малых коллективов исполнителей программного проекта;

сервисно-эксплуатационная деятельность:

- профилактическое и корректирующее сопровождение программного продукта в процессе эксплуатации;
- составление частного технического задания на разработку программного продукта.

1.3. Компетенции обучающегося, формируемые в результате освоения дисциплины

В результате освоения дисциплины у обучающихся формируются следующие компетенции:

Код компетенции	Наименование компетенции
1	2
ПК-8	Владение основами групповой динамики, психологии и профессионального поведения, специфичных для программной инженерии
ПК-10	Владение основными концепциями и моделями эволюции и сопровождения программного обеспечения

1.4. Перечень планируемых результатов обучения по дисциплине

Результаты освоения дисциплины	Технологии формирования компетенции по указанным результатам	Средства и технологии оценки по указанным результатам
Знает: - существующие современные модели, ключевые концепции и технологии групповой разработки программных систем (ПК-8) - существующие современные модели, ключевые концепции и технологии разработки программных систем (ПК-10)	Лекции, лабораторные занятия	Собеседование
Умеет: - использовать групповые подходы к инженерному проектированию в	Лекции, лабораторные занятия	Собеседование

конкретных предметных областях (ПК-8) - использовать различные подходы к инженерному проектированию в конкретных предметных областях (ПК-10)		
Имеет практический опыт: - разработки программной документации, командной разработки программного обеспечения (ПК-8) - разработки программной документации, персональной и командной разработки программного обеспечения, самостоятельного анализа новых тенденций и концепций программной инженерии (ПК-10)	Лекции, занятия лабораторные	Собеседование

2. Место дисциплины в структуре образовательной программы

Дисциплина относится к дисциплинам по выбору. Ее освоение осуществляется в 3 и 4 семестрах (заочная форма (февраль)); в 4 и 5 семестрах (очная и заочная формы).

№ п/п	Наименование дисциплин, определяющих междисциплинарные связи	Код компетенции(й)
1	Предшествующие дисциплины	
1.1	Информатика	ОПК-4
1.2	Теоретическая информатика	ОПК-1
2	Последующие дисциплины	
2.1	Проектирование и архитектура программных систем	ПК-1 ПК-5 ПК-7 ПК-10
2.2	Управление программными проектами	ПК-6 ПК-7 ПК-9 ПК-10

3. Объем дисциплины в зачетных единицах с указанием количества академических часов, выделенных на контактную работу обучающихся с преподавателем (по видам учебных занятий) и на самостоятельную работу

Распределение фонда времени по семестрам и видам занятий

Виды занятий	очная форма обучения	заочная форма обучения (февраль)	заочная форма обучения
Итого часов	144 ч.	144 ч.	144 ч.
Зачетных единиц	4 з.е.	4 з.е.	4 з.е.
Лекции (час)	18/12	2/2	2/2
Практические (семинарские) занятия (час)	-	-	-
Лабораторные работы (час)	14/18	2/6	2/6
Самостоятельная работа (час)	40/15	64/55	64/55

Курсовой проект (работа)	-	-	-
Контрольная работа	-	-	-
Экзамен, семестр /час.	5 сем./27	4 сем/9	5 сем/9
Зачет, семестр	4 сем.	3 сем./4 ч	4 сем./4 ч
Контрольная работа, семестр	-	-	-

4. Содержание дисциплины, структурированное по темам (разделам) с указанием отведенного на них количества академических часов и видов учебных занятий

4.1. Содержание дисциплины

№ п/п	Раздел дисциплины	Виды учебных занятий, включая самостоятельную работу студентов и трудоемкость (в академических часах)				Средства и технологии оценки
		Лекции, час	Практические (семинарские) занятия, час	Лабораторные работы, час	Самостоятельная работа, час	
	4/3/4 семестр					
1	Тема 1. Предмет изучения. Основное содержание 1. Программная инженерия. 2. Программное обеспечение.	2/1/1	-/-/-	-/-/-	2/2/2	Отчет по лабораторным работам
2	Тема 2. Процесс разработки программного обеспечения. Основное содержание 1. Процесс. 2. Совершенствование процесса. 3. Классические модели процесса.	2/1/1	-/-/-	6/2/2	4/8/8	Отчет по лабораторным работам
3	Тема 3. Рабочий продукт, дисциплина обязательств, проект. Основное содержание 1. Рабочий продукт. 2. Дисциплина обязательств. 3. Проект.	2/-/-	-/-/-	4/-/-	4/6/6	Отчет по лабораторным работам
4	Тема 4. Архитектура программного обеспечения. Основное содержание 1. Обсуждение. 2. Определение. 3. Множественность точек зрения. 4. Язык UML. 5. Виды диаграмм.	2/-/-	-/-/-	4/-/-	10/12 /12	Отчет по лабораторным работам
5	Тема 5. Управление требованиями. Основное содержание 1. Проблема. 2. Виды и свойства требований. 3. Варианты формализации требований. 4. Цикл работы с требованиями.	2/-/-	-/-/-	-/-/-	4/8/8	Отчет по лабораторным работам
6	Тема 6. Конфигурационное управление. Основное содержание 1. Проблема. 2. Единицы конфигурационного управления. 3. Управление версиями. 4. Управление сборками. 5. Понятие baseline. 6. Алгоритмы составления расписания. 7. Задача упаковки	2/-/-	-/-/-	-/-/-	4/8/8	Отчет по лабораторным работам

7	Тема 7. Тестирование. Основное содержание 1. Управление качеством. 2. Тестирование. 3. Работа с ошибками.	2/-/-	-/-/-	-/-/-	4/8/8	Отчет по лабораторным работам
8	Тема 8. Диаграммные техники в работе со знаниями. Основное содержание 1. Метод случаи использования. 2. Интерактивный цикл автор/рецензент. 3. Карты памяти.	2/-/-	-/-/-	-/-/-	4/8/8	Отчет по лабораторным работам
9	Тема 9. MSF. Основное содержание 1. История и текущий статус. 2. Основные принципы. 3. Модель команды. 4. Прочие особенности.	2/-/-	-/-/-	-/-/-	4/4/4	Отчет по лабораторным работам
	Промежуточная аттестация по дисциплине	18/2/2	-/-/-	14/2/2	40/64/64	Зачет
5/4/5 семестр						
10	Тема 10. CMMI. Основное содержание 1. Что такое CMMI? 2. Уровни зрелости процессов по CMMI.	2/-/-	-/-/-	-/-/-	1/5/5	Отчет по лабораторным работам
11	Тема 11. "Гибкие" (agile) методы разработки. Основное содержание 1. Общие вопросы. 2. Extreme Programming. 3. Scrum. 4. Полярный метод генерации случайных чисел с нормальным распределением.	2/-/-	-/-/-	-/-/-	1/5/5	Отчет по лабораторным работам
12	Тема 12. Обзор технологии Microsoft Visual Studio Team System (VSTS). Основное содержание 1. Общий обзор. 2. Состав продукта. 3. Правила инсталляции. 4. Пакет Team Explorer.	2/1/1	-/-/-	4/4/4	4/12/12	Отчет по лабораторным работам
13	Тема 13. VSTS: управление элементами работ (Work Items). Основное содержание 1. Определение, свойства, жизненный цикл. 2. Средства использования.	2/-/-	-/-/-	4/-/-	2/8/8	Отчет по лабораторным работам
14	Тема 14. VSTS: конфигурационное управление. Основное содержание 1. Система контроля версий. 2. Автоматические сборки	2/-/-	-/-/-	6/2/2	4/12/12	Отчет по лабораторным работам
15	Тема 15. VSTS: тестирование. Основное содержание 1. Система отслеживания ошибок. 2. Модульные тесты. 3. Пакеты тестов. 4. Автоматическое тестирование Web-приложений.	1/1/1	-/-/-	2/-/-	2/8/8	Отчет по лабораторным работам
16	Тема 16. VSTS: поддержка различных моделей процесса. Основное содержание 1. Поддержка шаблонов процесса. 2. Обзор существующих шаблонов.	1/-/-	-/-/-	2/-/-	1/5/5	Отчет по лабораторным работам

	3. MSF for Agile Software Development. 4. Scrum.					
	Промежуточная аттестация по дисциплине	12/2/2	-/-/-	18/6/6	15/55/55	Экзамен

Примечание:

–/–/–, объем часов соответственно для очной формы обучения, заочной формы обучения (февраль), заочной формы обучения

4.2.Содержание практических (семинарских) занятий

Практические и семинарские занятия учебным планом не предусмотрены

4.3.Содержание лабораторных работ

№	Наименование лабораторных работ	Объем часов	Наименование темы дисциплины
	4/3/4 семестр		
1	Лабораторная работа 1. «Основные элементы определения, представления, проектирования и моделирования программных систем с помощью языка UML.»	4/-/-	Тема 2. Процесс разработки программного обеспечения
2	Лабораторная работа 2. «Методология функционального моделирования»	2/2/2	Тема 2. Процесс разработки программного обеспечения
3	Лабораторная работа 3 «. Разработка описания и анализ информационной системы»	2/-/-	Тема 3. Рабочий продукт, дисциплина обязательств, проект.
4	Лабораторная работа 4. Разработка требований к информационной системе	2/-/-	Тема 3. Рабочий продукт, дисциплина обязательств, проект.
5	Лабораторная работа 5 Методология управления проектами	4/-/-	Тема 4. Архитектура программного обеспечения
	Итого за 4/3/4 семестр	14/2/2	
	5/4/5 семестр		
1	Лабораторная работа 6. Знакомство с условиями модельной задачи, настройка инфраструктуры TFS для будущей разработки	2/2/2	Тема 12. Обзор технологии Microsoft Visual Studio Team System (VSTS).
2	Лабораторная работа 7. Работа с системой отслеживания ошибок	2/2/2	Тема 12. Обзор технологии Microsoft Visual Studio Team System (VSTS).
3	Лабораторная работа 8. Работа с системой контроля версий	2/-/-	Тема 13. VSTS: управление элементами работ (Work Items).
4	Лабораторная работа 9. Разработка модульных тестов	2/-/-	Тема 13. VSTS: управление элементами работ (Work Items).
5	Лабораторная работа 10. Создание и конфигурация автоматической сборки	4/-/-	Тема 14. VSTS: конфигурационное управление.
6	Лабораторная работа 11. Настройка шаблона процесса	2/2/2	Тема 14. VSTS: конфигурационное управление.
7	Лабораторная работа 12. Основные этапы работы в Microsoft Visual Studio Team System (VSTS)	2/-/-	Тема 15. VSTS: тестирование.
8	Лабораторная работа 13. Основные элементы среды разработки Visual Studio Integrated Development Environment (IDE - интегрированная среда разработки) C# при создании на языке C# приложений с графически интерфейсом	2/-/-	Тема 16. VSTS: поддержка различных моделей процесса.

	Итого за 5/4/5 семестр	18/6/6	
	Итого	32/8/8	

Примечание:

–/–/–, объем часов соответственно для очной формы обучения, заочной формы обучения (февраль), заочной формы обучения

5. Учебно-методическое обеспечение самостоятельной работы обучающихся по дисциплине

Технологическая карта самостоятельной работы студента

Код реализующей компетенции	Вид деятельности студентов (задания на самостоятельную работу)	Итоговый продукт самостоятельной работы	Средства и технологии оценки	Объем часов
1	2	3	4	5
ПК-8	Выполнение индивидуальных заданий в виде реферата или конспекта на заданную тему	реферат или конспект	собеседование	20/32/32
ПК-10	Выполнение индивидуальных заданий в виде реферата или конспекта на заданную тему	реферат или конспект	собеседование	20/32/32
Итого за 4/3/4 семестр				40/64/64
ПК-8	Выполнение индивидуальных заданий в виде реферата или конспекта на заданную тему	реферат или конспект	собеседование	5/30/30
ПК-10	Выполнение индивидуальных заданий в виде реферата или конспекта на заданную тему	реферат или конспект	собеседование	10/25/25
Итого за 5/4/5 семестр				15/55/55
Итого				55/119/119

Примечание:

–/–/–, объем часов соответственно для очной формы обучения, заочной формы обучения (февраль), заочной формы обучения

Литература:

- Гагарина, Л. Г. Введение в архитектуру программного обеспечения [Электронный ресурс] : учеб. пособие по направлениям подгот. 09.03.04 "Прогр. инженерия" (профиль бакалавриата "Прогр. технологии распредел. обраб. информ."), 09.04.04 "Прогр. инженерия" (прогр. магистратуры "Прогр. обеспечение автоматизир. систем и вычисл. комплексов") / Л. Г. Гагарина, А. Р. Федоров, П. А. Федоров. - Документ Bookread2. - М. : ФОРУМ [и др.], 2017. - Режим доступа: <http://znaniyum.com/bookread2.php?book=615207#>
- Гагарина, Л. Г. Технология разработки программного обеспечения [Электронный ресурс] : учеб. пособие для вузов по направлениям подгот. 09.04.01 и 09.03.03 "Информатика и вычисл. техника" / Л. Г. Гагарина, Е. В. Кокорева, Б. Д. Виснадул ; под ред. Л. Г. Гагариной. - Документ Bookread2. - М. : Форум [и др.], 2017. - 399 с. - Режим доступа: <http://znaniyum.com/bookread2.php?book=768473>

Содержание заданий для самостоятельной работы

Темы рефератов (письменных работ, эссе, докладов и т.п.)

1. Лауреаты премии Тьюринга (по годам).
2. Эволюция сложных программных систем.
3. Методы документирования архитектуры.
4. Управление знаниями в процессе разработки программных систем.
5. CASE технологии разработки программных систем.
6. Модели программных систем.
7. Построение процесса разработки программных систем.
8. Бизнес аспекты разработки программных систем.
9. Модели ROI для оценки эффективности компаний-разработчиков программного обеспечения.
10. Человеческий фактор при разработке ПО.
11. Модели и методы оценки личностных характеристик исполнителей и команды в целом.
12. Оценка затрат программных проектов методом функциональных точек.
13. Регрессионная модель оценки затрат программных проектов COSOMO II.
14. Оценка программных проектов в модели SLIM.
15. Методы выбора организационной формы реализации программного проекта.
16. Количественные методики оценки рисков программных проектов.
17. Метрические показатели в оценке программных проектов.
18. Модели структурного анализа программных проектов.
19. Модели объектно-ориентированного анализа программных проектов.
20. Метод определения точек тестирования, основанный на анализе цикломатической сложности Мак-Кейба.
21. Сравнительный анализ инструментов моделирования и трассировки программных требований.
22. Сравнительный анализ инструментов верификации программных проектов.
23. Сравнительный анализ инструментов оптимизации программных проектов.
24. Сравнительный анализ инструментов тестирования программного обеспечения (генераторы тестов, схемы выполнения тестов, оценка тестов, управление тестами).
25. Сравнительный анализ инструментов сопровождения программного обеспечения.
26. Системы моделирования процессов разработки программного обеспечения.
27. Среды разработки программного обеспечения, ориентированные на процессы.
28. Сравнительный анализ инструментов обеспечения качества программного обеспечения.
29. Сравнительный анализ инструментов управления конфигурацией программного обеспечения.
30. Инструменты планирования и отслеживания программных проектов.
31. Инструменты, реализующие поддержку инфраструктуры разработки.

Вопросы (тест) для самоконтроля

1. В каких пределах возможна идентификация реквизита с помощью ссылок?
2. В каких режимах реквизит может участвовать в отчетах?
3. В каких случаях ошибка может попасть в состояние Active?
4. В каких случаях ошибка может попасть в состояние Closed?
5. В каких случаях ошибка может попасть в состояние Resolved?
6. В какой модели каждый виток представляет собой фазу разработки?
7. В какой практике Scrum может участвовать представитель заказчика?

8. В каком виде могут быть представлены требования?
9. В каком случае возврат инвестиций от внедрения происходит быстрее?
10. В каком случае приведен пример использования стратегии organization pull?
11. В каком случае приведен пример использования стратегии technology push?
12. В каком случае речь идет о бизнес-реинжиниринге?
13. В результате какого вида деятельности требования должны быть оформлены в виде структурированного набора документов и моделей?
14. В чем заключается "плоскость" модели MSF?
15. В чем заключается основное отличительное свойство системы контроля версий в TFS?
16. В чем заключается принцип гибкости в методологии MSF?
17. В чем заключается согласованность ПО?
18. В чем отличие информатики от программной инженерии?
19. Для какой роли отсутствует отдельное издание Visual Studio?
20. Для решения каких задач применяются инструменты тестирования?
21. Для чего используется команда Branch?
22. Для чего используется команда Merge?
23. Для чего используется CMMI?
24. Для чего используется рабочий продукт?
25. Для чего осуществляется рефакторинг кода?
26. Для чего осуществляется совершенствование процесса разработки ПО?
27. Для чего предназначена команда Publish?
28. Для чего предназначены диаграммы компонент?
29. Для чего предназначены диаграммы конечных автоматов?
30. Для чего предназначены диаграммы случаев использования?
31. Для чего применяются варианты использования (Use Case)?
32. Из скольких уровней состоит TFS?
33. К информатике относятся:...
34. К какому типу проектов относятся проекты по разработке ПО?
35. Какая из указанных моделей является двумерной?
36. Какая из указанных практик проводится в конце каждого Sprint?
37. Какая из указанных практик проводится в начале каждого Sprint?
38. Какая команда предназначена для внесения изменений в VSTS?
39. Какая команда предназначена для получения обновлений с VSTS?
40. Какая область объединяет различные инженерные дисциплины по разработке всевозможных искусственных систем?
41. Какая система используется для хранения всех основных артефактов, составляющих результат деятельности проектной команды?
42. Какая система позволит автоматизировать получение образа конечного продукта в виде, готовом для тестирования и отправки заказчику?
43. Какая стратегия нацелена на решение конкретных проблем компании?
44. Какие виды деятельности включает в себя процесс создания ПО?
45. Какие виды деятельности входят в состав программной инженерии?
46. Какие возвраты возможны при разработке по водопадной модели?
47. Какие возвраты невозможны при разработке по водопадной модели?
48. Какие вопросы решаются в сфере программной инженерии?
49. Какие группы областей усовершенствования выделяются в CMMI?
50. Какие диаграммы используются для моделирования аппаратной части системы, с которой связано ПО?
51. Какие диаграммы используются для спецификации бизнес-процессов, которые должно автоматизировать разрабатываемое ПО?
52. Какие диаграммы относятся к поведенческим диаграммам?
53. Какие диаграммы относятся к структурным диаграммам?
54. Какие диаграммы относятся к структурным диаграммам?

55. Какие задачи решает комитет технического контроля?
56. Какие задачи решает комитет технического контроля?
57. Какие из перечисленных принципов справедливы для организации процесса по XP (Extreme Programming)?
58. Какие из перечисленных принципов справедливы для организации процесса по XP (Extreme Programming)?
59. Какие из приведенных примеров не являются единицами конфигурационного управления?
60. Какие из приведенных примеров являются единицами конфигурационного управления?
61. Какие из указанных методов обеспечения качества ПО относятся к формальным методам?
62. Какие методологии могут использоваться для реализации CMMI?
63. Какие продукты называются единицами конфигурационного управления?
64. Какие продукты не относятся к серверной части VSTS?
65. Какие продукты относятся к клиентской части VSTS?
66. Какие продукты относятся к серверной части VSTS?
67. Какие роли имеются в цикле "читатель/автор"?
68. Какие специалисты вовлечены в разработку и/или использование ПО?
69. Какие средства входят в VSTS?
70. Какие средства предоставляет инструмент Microsoft Visual Studio Team System 2008 Development Edition?
71. Какие средства предоставляет инструмент Microsoft Visual Studio Team System 2008 Architecture Edition?
72. Какие средства предоставляет инструмент Microsoft Visual Studio Team System 2008 Test Edition?
73. Какие структурные диаграммы используются для описания связей классов друг с другом?
74. Какие тесты могут быть воспроизведены без участия человека?
75. Какие тесты представляют собой последовательность действий тестировщика или разработчика, приводящую к воспроизведению ошибки?
76. Какие требования описывает такие характеристики системы, как надежность, особенности поставки, определенный уровень качества?
77. Какие требования определяют то, что система должна уметь делать?
78. Какие утверждения верны для водопадной модели?
79. Какие утверждения верны для методологии MSF?
80. Какие участники имеют право принимать решения на совещаниях Daily Scrum Meeting в Scrum?
81. Какие участники имеют право принимать участие в совещаниях Daily Scrum Meeting в Scrum?
82. Какие функции фиксируются при составлении вариантов использования?
83. Какие характеристики системы определяются нефункциональными требованиями?
84. Каким образом могут осуществляться переходы между состояниями?
85. Какими системными реквизитами определяется жизненный цикл элемента работы?
86. Какова основная задача комитета ITU?
87. Какова основная задача организации ETSI?
88. Какова основная задача организации ISO?
89. Какова основная задача удовлетворения потребителя?
90. Какова основная задача управления выпуском?
91. Какова основная задача управления программой?
92. Какова основная задача управления продуктом?
93. Какова последовательность действий при составлении Use Case-диаграмм?
94. Какова цель создания прототипа при использовании водопадной модели?
95. Каково назначение продукта Share Point?
96. Каково назначение продукта SQL Server?
97. Каково назначение продукта Team Explorer?
98. Каково назначение пятнадцатиминутных ежедневных совещаний (Daily Scrum Meeting) в Scrum?
99. Каковы достоинства водопадной модели?
100. Каковы недостатки водопадной модели?
101. Каковы основные атрибуты ошибок?

102. Какое действие необходимо выполнить, чтобы ветвь попала на сервер?
103. Какое свойство зависит от размера программных объектов?
104. Какое свойство обозначает однозначность понимания требований заказчиком и разработчиками?
105. Какое свойство определяет процедуры внесения изменений в требования?
106. Какой вид деятельности процесса разработки ПО акцентирует внимание на принципах реализации ПО?
107. Какой вид деятельности решает задачу оценки понятности сформулированных требований и их характеристик?
108. Какой из указанных элементов работы имеет набор различных состояний, перечень событий, изменяющих эти состояния, а также ответственное лицо?
109. Какой из участников создания модели при описании системы не несет ответственности за качество моделирования?
110. Какой из участников создания модели при описании системы несет ответственность за качество моделирования?
111. Какой комитет занимается разработкой стандартов по радиоэлектронике и электротехнике?
112. Какой комитет занимается разработкой и продвижением объектно-ориентированных технологий и стандартов?
113. Какой комитет занимается стандартизацией в области программной инженерии, выработкой критериев для сертификации надежных и зрелых компаний?
114. Какой процесс разработки ПО является универсальным для разработки ПО любого вида?
115. Какой ролевой кластер отвечает за высокоуровневое проектирование решения и создание функциональной спецификации ПО?
116. Какой системный реквизит описывает переходы между состояниями?
117. Какой системный реквизит описывает текущее состояние элемента работы и определяет его текущую роль в процессе?
118. Какой способ позволяет формализовать требования?
119. Какой уровень TFS реализован на основе продуктов MS SQL Server и Share Point?
120. Какой уровень TFS реализован на основе технологии ASP.NET?
121. Какой этап непосредственно предшествует выполнению итерации при использовании метода Scrum?
122. Какой этап следует за анализом результатов и пересмотром требований при использовании метода Scrum?
123. Какой этап следует за созданием требований к продукту при использовании метода Scrum?
124. Какую роль выполняет Scrum-команда?
125. Какую роль выполняет Scrum-мастер?
126. Какую роль выполняет владелец продукта?
127. Какую роль выполняет менеджер в процессе работы над ошибками?
128. Когда осуществляется синхронизация с менеджером и заказчиком при использовании метода Scrum?
129. Кто может быть актером в Use Case-диаграммах?
130. На взаимозависимости каких показателей основано управление компромиссами?
131. На какой стадии разработки применяют случаи использования?
132. На каком уровне зрелости осуществляется анализ причин возникновения проблем и предотвращение их появления в будущем?
133. На каком уровне зрелости осуществляется управление рисками?
134. На каком уровне процессы в полной мере существуют лишь в рамках отдельных проектов?
135. На каком уровне существует стандартный процесс в рамках всей компании в целом?
136. На каком уровне, по определению, находится любая компания?
137. От чего зависит формализация требований в проекте?
138. Отметьте артефакты процесса разработки ПО:
139. Отметьте верные утверждения:
140. Отметьте верные утверждения:
141. Отметьте верные утверждения:
142. Отметьте верные утверждения:

143. При выполнении какого вида тестирования две и более компонент тестируются на совместимость?
144. При выполнении какого вида тестирования система тестируется на устойчивость к непредвиденным ситуациям?
145. При выполнении какого вида тестирования тестируется вся система в целом?
146. При выполнении какого вида тестирования тестируется отдельный модуль, в отрыве от остальной системы?
147. При использовании какого метода тестирования код программы доступен тестировщикам?
148. При использовании какого метода тестирования реализация системы недоступна тестировщикам?
149. При использовании какой стратегии изменения, вносимые в процесс, более глобальны?
150. С какими участниками взаимодействует автор при создании модели системы?
151. С какой ролью можно совмещать разработку?
152. С какой ролью нельзя совмещать тестирование?
153. Укажите отличия программной инженерии от программирования?
154. Целью какого вида деятельности является обнаружение и устранение противоречий и неоднозначностей в требованиях, их уточнение и систематизация?
155. Чем определяется набор реквизитов элемента работы?
156. Чем определяется сложность ПО?
157. Что включает в себя архитектура ПО?
158. Что входит в состав ИТ-решения?
159. Что входит в состав систем контроля ошибок?
160. Что может являться рабочим продуктом в процессе разработки ПО?
161. Что обозначает состояние ошибки Active?
162. Что обозначает состояние ошибки Closed?
163. Что обозначает состояние ошибки Resolved?
164. Что обозначает ясность и недвусмысленность требований?
165. Что описывают правила в жизненном цикле реквизита?
166. Что определяют варианты использования?
167. Что реализуют модели, представленные диаграммами UML?
168. Что такое бизнес-реинжиниринг?
169. Что такое вид деятельности?
170. Что такое интеграционное тестирование?
171. Что такое информатика?
172. Что такое конфигурационное управление?
173. Что такое критерии тестирования?
174. Что такое нагрузочное тестирование?
175. Что такое регрессионное тестирование?
176. Что такое управление версиями?
177. Что такое управление сборками?
178. Что такое фаза разработки?
179. Что является причиной создания и использования различных моделей разработки ПО, выполненных с разных точек зрения?
180. Что является рабочим продуктом в проекте?
181. Что является управляющей информацией в проекте?
182. Что является характеристикой точки зрения?

6. Методические указания для обучающихся по освоению дисциплины Инновационные образовательные технологии

Вид образовательных технологий, средств передачи знаний, формирования умений и	№ темы / тема лекции	№ практического занятия/наименование темы	№ лабораторной работы / цель
--	----------------------	---	------------------------------

практического опыта			
Слайд-лекции	Тема 1. Предмет изучения.		
Слайд-лекции	Тема 2. Процесс разработки программного обеспечения.		
Слайд-лекции	Тема 3. Рабочий продукт, дисциплина обязательств, проект.		

В начале семестра студентам необходимо ознакомиться с технологической картой дисциплины, выяснить, какие результаты освоения дисциплины заявлены (знания, умения, практический опыт). Для успешного освоения дисциплины студентам необходимо выполнить задания, предусмотренные рабочей учебной программой дисциплины и пройти контрольные точки в сроки, указанные в технологической карте (раздел 11). От качества и полноты их выполнения будет зависеть уровень сформированности компетенции и оценка текущей успеваемости по дисциплине. По итогам текущей успеваемости студенту может быть выставлена оценка по промежуточной аттестации, если это предусмотрено технологической картой дисциплины. Списки учебных пособий, научных трудов, которые студентам следует прочесть и законспектировать, темы лабораторных работ и вопросы к ним, вопросы к экзамену (зачету) и другие необходимые материалы указаны в разработанном для данной дисциплины учебно-методическом комплексе.

Основной формой освоения дисциплины является контактная работа с преподавателем - лекции, лабораторные работы, консультации (в том числе индивидуальные), в том числе проводимые с применением дистанционных технологий.

По дисциплине часть тем (разделов) изучается студентами самостоятельно. Самостоятельная работа предусматривает подготовку к аудиторным занятиям, выполнение заданий (письменных работ, творческих проектов и др.) подготовку к промежуточной аттестации (экзамену (зачету)).

На лекционных занятиях и лабораторных работах вырабатываются навыки и умения обучающихся по применению полученных знаний в конкретных ситуациях, связанных с будущей профессиональной деятельностью. По окончании изучения дисциплины проводится промежуточная аттестация (экзамен, зачет)).

Регулярное посещение аудиторных занятий не только способствует успешному овладению знаниями, но и помогает организовать время, т.к. все виды учебных занятий распределены в семестре планомерно, с учетом необходимых временных затрат.

6.1. Методические указания для обучающихся по освоению дисциплины на лабораторных работах

Лабораторные работы

№	Наименование лабораторных работ	Задание по лабораторным работам
1	Лабораторная работа 1. «Основные элементы определения, представления, проектирования и моделирования программных систем с помощью языка UML.»	1. Изучить теоретический материал. 2. Построить диаграмму вариантов использования для выбранной информационной системы. 3. Выполнить реализацию вариантов использования в терминах взаимодействующих объектов и представляющую собой набор диаграмм: (диаграммы классов, диаграммы взаимодействия). 4. Разделить классы по пакетам используя один из механизмов разбиения. 5. Построить диаграмму состояний для конкретных объектов информационной системы.
2	Лабораторная работа 2.	1. Изучить теоретический материал.

	«Методология функционального моделирования»	2. Построить функциональную модель системы, описанной в Лабораторной работе № 1 так, чтобы она отвечала всем предъявленным к системе требованиям, представляла полный функционал системы и её основные бизнес-процессы: • с помощью методологии IDEF0 построить контекстную диаграмму; • с помощью методологии IDEF0 построить диаграмму 1-го уровня (A0) – модель окружения; • с помощью методологии IDEF3 декомпозировать функциональные блоки модели окружения на 1-2 уровня вглубь до потоков, связи с внешними системами и • на каждой диаграмме 2-го уровня должно быть не менее 4-х функциональных блоков; • на каждой диаграмме 3-го уровня и далее не менее 2-х функциональных блоков.
3	Лабораторная работа 3 «. Разработка описания и анализ информационной системы»	1. Изучить теоретический материал. 2. Составить подробное описание информационной системы. 3. На основании описания системы провести анализ осуществимости. 4. Распределить роли в группе (руководитель проекта-разработчик, системный аналитик-разработчик, тестер-разработчик). 5. Заполнить разделы плана: • Введение • Организация выполнения проекта • Анализ рисков 6. Разделы должны содержать рекомендации относительно разработки системы, базовые предложения по объёму требуемого бюджета, числу разработчиков, времени и требуемому программному обеспечению.
4	Лабораторная работа №4. Разработка требований к информационной системе	1. Изучить теоретический материал. 2. Построить опорные точки зрения на основании метода VORD для формирования и анализа требований. Результатом должны явиться две диаграммы: диаграмма идентификации точек зрения и диаграмма иерархии точек зрения. 3. Составить информационную модель будущей системы, включающую в себя описание основных объектов системы и взаимодействия между ними. На основании полученной информационной модели и диаграмм идентификации точек зрения, диаграмма иерархии точек зрения сформировать требования пользователя и системные требования. 4. Провести аттестацию требований, указать какие типы проверок выбрали. 5. На основании описания системы (Лабораторная работа №3), информационной модели, пользовательских и системных требований составить техническое задание на создание программного обеспечения. ТЗ должно содержать основные разделы, описанные в ГОСТ 34.602-89.
5	Лабораторная работа №5 Методология управление проектами	1. Изучить предлагаемый теоретический материал. 2. Построить временную и сетевую диаграммы для выбранного проекта. 3. Построить диаграмму распределения участников группы

		по этапам. 4. Построить список возможных рисков с указанием названия риска, его описание и типа. 5. Провести анализ рисков. 6. Описать стратегию планирования рисков.
6	Лабораторная работа №6. Знакомство с условиями модельной задачи, настройка инфраструктуры TFS для будущей разработки	1. Организовать команду студентов. 2. Ознакомиться с условиями модельной задачи 3. Настроить инфраструктуру TFS для будущей разработки (создать командный проект и распределить права на работу с ним)
7	Лабораторная работа №7. Работа с системой отслеживания ошибок	1. Произвести планирование работ на основе методологии Scrum 2. Изучить способы использования системы отслеживания задач TFS. 3. Импортировать список пользовательских историй файла Excel в TFS 4. Спланировать будущий спринт и распределить задачи.
8	Лабораторная работа №8. Работа с системой контроля версий	1. Реализовать решение модельной задачи. 2. Освоить систему контроля версий TFS, её интеграцию с системой отслеживания задач 3. Создать ветви и интеграции изменений
9	Лабораторная работа №9. Разработка модульных тестов	1. Использовать средства автоматической генерации тестов (по шагам в теоретическом материале) 2. Наполнить сгенерированные тесты содержимым 3. Изменить конфигурацию запуска модульных тестов и считать тестовое покрытие.
10	Лабораторная работа №10. Создание и конфигурация автоматической сборки	1. Создать несколько определений для автоматической сборки (build definitions) в разных случаях (с тестами и без, с анализом кода и без и т.д.). 2. Настроить параметры непрерывной интеграции и рассылки уведомлений.
11	Лабораторная работа №11. Настройка шаблона процесса	1. Провести ретроспективный анализ выполненного Scrum sprint. 2. Выявить потенциальные способы оптимизации и применить их, используя средства настройки процесса разработки TFS. 3. Изменить настройки системы отслеживания задач средствами Team Foundation Server Power Tools.
12	Лабораторная работа №12. Основные этапы работы в Microsoft Visual Studio Team System (VSTS)	1. Изучить состав Visual Studio 2008 Team Suite и четыре ролевые версии: для архитекторов — Architecture Edition, для разработчиков — Development Edition, для разработчиков баз данных — Database Edition и для тестировщиков Test Edition.
13	Лабораторная работа №13. Основные элементы среды разработки Visual Studio Integrated Development Environment (IDE - интегрированная среда разработки) C# при создании на языке C# приложений с графически интерфейсом	1. Изучить теоретический материал. 2. Создать Windows форму. 3. Добавить в форму две кнопки (1 и 2), для которых задать соответственно цвета Красный и Синий (свойство BackColor). 4. Написать для кнопок 1 и 2 обработчики, которые изменяют цвета кнопок: при неоднократном нажатии любой кнопки цвета кнопок меняются (цвет кнопки 1 меняется на цвет кнопки 2 и наоборот). 5. Протестировать работу приложения 6. На Windows форме создать кнопку "Решение".

		7. Написать для кнопки «Решение» обработчик, который выдает сообщение с результатом решения линейной задачи согласно варианту: 8. Добавьте кнопку "Выход". Закрытие приложения обеспечивает метод Exit() класса Application. 9. Протестировать работу приложения.
--	--	---

Лабораторные работы обеспечивают:

формирование умений и навыков обращения с приборами и другим оборудованием, демонстрацию применения теоретических знаний на практике, закрепление и углубление теоретических знаний, контроль знаний и умений в формулировании выводов, развитие интереса к изучаемой дисциплине.

Применение лабораторных работ позволяет вовлечь в активную работу всех обучающихся группы и сформировать интерес к изучению дисциплины.

Самостоятельный поиск ответов на поставленные вопросы и задачи в ходе лабораторной работы приобретают особую значимость в восприятии, понимании содержания дисциплины.

Изученный на лекциях материал лучше усваивается, лабораторные работы демонстрируют практическое их применение.

6.2 Методические указания для выполнения контрольных работ

Контрольная работа учебным планом не предусмотрена.

6.3 Методические указания для выполнения курсовых работ (проектов)

Курсовая работа учебным планом не предусмотрена

7. Фонд оценочных средств для проведения текущего контроля успеваемости и промежуточной аттестации обучающихся по дисциплине (экзамен, зачет)

Фонды оценочных средств, позволяющие оценить уровень сформированности компетенций и результаты освоения дисциплины, представлены следующими компонентами:

Код оцениваемой компетенции (или ее части)	Тип контроля	Вид контроля	Количество элементов, <i>шт.</i>
ПК-8	текущий	устный опрос	1-50
ПК-8	промежуточный	тест	1-80
ПК-10	текущий	устный опрос	81-130
ПК-10	промежуточный	тест	81-160

7.1 Оценочные средства для текущего контроля успеваемости, промежуточной аттестации по итогам освоения дисциплины

Результаты освоения дисциплины	Оценочные средства
Знает: - существующие современные модели, ключевые концепции и технологии групповой разработки программных систем (ПК-8) - существующие современные модели, ключевые концепции и	- Выберите набор свойств программного обеспечения: - сложность, согласованность, изменяемость, нематериальность; - сложность, согласованность, изменяемость, материальность; - сложность, согласованность, изменяемость, виртуальность; - К свойствам требований к разработке ПО не относится: - ясность

технологии разработки программных систем (ПК-10)	б) недвусмысленность в) лаконичность
Умеет: - использовать групповые подходы к инженерному проектированию в конкретных предметных областях (ПК-8) - использовать различные подходы к инженерному проектированию в конкретных предметных областях (ПК-10)	3. Укажите правильное сочетание обозначений Актеров (в стандарте UML): а) типовые пользователи ПО; выделенный пользователь б) типовые пользователи ПО; другие системы, взаимодействующие с данной; выделенный пользователь в) типовые пользователи ПО; другие системы, взаимодействующие с данной 4. В технике MSF используется: а) водопадная модель б) спиральная модель в) сочетание водопадной и спиральной моделей
Имеет практический опыт: - разработки программной документации, командной разработки программного обеспечения (ПК-8) - разработки программной документации, персональной и командной разработки программного обеспечения, самостоятельного анализа новых тенденций и концепций программной инженерии (ПК-10)	Описание требований к разработке ПО для разработки и тестирования сложной системы. Разработка структуры сложной программной системы для выполнения задач организаций и учреждений

7.2. Методические рекомендации к определению процедуры оценивания знаний, умений, навыков и (или) опыта деятельности

Рабочая учебная программа дисциплины содержит следующие структурные элементы:

- перечень компетенций, формируемых в результате изучения дисциплины с указанием этапов их формирования в процессе освоения образовательной программы;
- типовые контрольные задания или иные материалы, необходимые для оценки знаний, умений, навыков и (или) опыта деятельности, характеризующих этапы формирования компетенций в процессе освоения образовательной программы (далее—задания). Задания по каждой компетенции, как правило, не должны повторяться.

Требования по формированию задания на оценку ЗНАНИЙ:

- обучающийся должен воспроизводить и объяснять учебный материал с требуемой степенью научной точности и полноты;
- применяются средства оценивания компетенций: тестирование, вопросы по основным понятиям дисциплины и т.п.

Требования по формированию задания на оценку УМЕНИЙ:

- обучающийся должен решать типовые задачи (выполнять задания) на основе воспроизведения стандартных алгоритмов решения;
- применяются следующие средства оценивания компетенций: простые ситуационные задачи (задания) с коротким ответом или простым действием, упражнения, задания на соответствие или на установление правильной последовательности, эссе и другое.

Требования по формированию задания на оценку навыков и (или) ОПЫТА ДЕЯТЕЛЬНОСТИ:

- обучающийся должен решать усложненные задачи (выполнять задания) на основе приобретенных знаний, умений и навыков, с их применением в определенных ситуациях;
- применяются средства оценивания компетенций: задания требующие многошаговых решений как в известной, так и в нестандартной ситуациях, задания, требующие поэтапного решения и развернутого ответа, ситуационные задачи, проектная деятельность, задания расчетно-

графического типа. Средства оценивания компетенций выбираются в соответствии с заявленными результатами обучения по дисциплине.

Процедура выставления оценки доводится до сведения обучающихся в течение месяца с начала изучения дисциплины путем ознакомления их с технологической картой дисциплины, которая является неотъемлемой частью рабочей учебной программы по дисциплине.

В результате оценивания компетенций на различных этапах их формирования по дисциплине студенту начисляются баллы по шкале, указанной в рабочей учебной программе по дисциплине.

7.3. Описание показателей и критериев оценивания компетенций на различных этапах их формирования, описание шкал оценивания

Успешность усвоения дисциплины характеризуется качественной оценкой на основе листа оценки сформированности компетенций, который является приложением к зачетно-экзаменационной ведомости при проведении промежуточной аттестации по дисциплине.

Критерии оценивания компетенций

Компетенция считается сформированной, если теоретическое содержание курса освоено полностью; при устных собеседованиях студент исчерпывающе, последовательно, четко и логически стройно излагает учебный материал; свободно справляется с задачами, вопросами и другими видами заданий, требующих применения знаний, использует в ответе дополнительный материал; все предусмотренные рабочей учебной программой задания выполнены в соответствии с установленными требованиями, студент способен анализировать полученные результаты; проявляет самостоятельность при выполнении заданий, качество их выполнения оценено числом баллов от 86 до 100, что соответствует *повышенному уровню* сформированности компетенции.

Компетенция считается сформированной, если теоретическое содержание курса освоено полностью; при устных собеседованиях студент последовательно, четко и логически стройно излагает учебный материал; справляется с задачами, вопросами и другими видами заданий, требующих применения знаний; все предусмотренные рабочей учебной программой задания выполнены в соответствии с установленными требованиями, студент способен анализировать полученные результаты; проявляет самостоятельность при выполнении заданий, качество их выполнения оценено числом баллов от 61 до 85,9, что соответствует *пороговому уровню* сформированности компетенции.

Компетенция считается несформированной, если студент при выполнении заданий не демонстрирует знаний учебного материала, допускает ошибки, неуверенно, с большими затруднениями выполняет практические работы, не демонстрирует необходимых умений, доля невыполненных заданий, предусмотренных рабочей учебной программой составляет 55 %, качество выполненных заданий не соответствует установленным требованиям, качество их выполнения оценено числом баллов ниже 61, что соответствует *допороговому уровню*.

Шкала оценки уровня освоения дисциплины

Качественная оценка может быть выражена: в процентном отношении качества усвоения дисциплины, которая соответствует баллам, и переводится в уровневую шкалу и оценки «отлично» / 5, «хорошо» / 4, «удовлетворительно» / 3, «неудовлетворительно» / 2, «зачтено», «не зачтено». Преподаватель ведет письменный учет текущей успеваемости студента в соответствии с технологической картой по дисциплине.

Шкала оценки результатов освоения дисциплины, сформированности компетенций

Шкалы оценки уровня сформированности компетенции (й)		Шкала оценки уровня освоения дисциплины		
<i>Уровневая шкала оценки компетенций</i>	<i>100 балльная шкала, %</i>	<i>100 балльная шкала, %</i>	<i>5-балльная шкала, дифференцированная оценка/балл</i>	<i>недифференцированная оценка</i>
допороговый	ниже 61	ниже 61	«неудовлетворительно» / 2	не зачтено
пороговый	61-85,9	70-85,9	«хорошо» / 4	зачтено

		61-69,9	«удовлетворительно» / 3	зачтено
повышенный	86-100	86-100	«отлично» / 5	зачтено

8. Учебно-методическое и информационное обеспечение дисциплины

8.1 Перечень основной и дополнительной учебной литературы, необходимой для освоения дисциплины

Списки основной литературы

1. Гагарина, Л. Г. Введение в архитектуру программного обеспечения [Электронный ресурс] : учеб. пособие по направлениям подгот. 09.03.04 "Прогр. инженерия" (профиль бакалавриата "Прогр. технологии распредел. обраб. информ."), 09.04.04 "Прогр. инженерия" (прогр. магистратуры "Прогр. обеспечение автоматизир. систем и вычисл. комплексов") / Л. Г. Гагарина, А. Р. Федоров, П. А. Федоров. - Документ Bookread2. - М. : ФОРУМ [и др.], 2017. - Режим доступа: <http://znanium.com/bookread2.php?book=615207#>
2. Гагарина, Л. Г. Технология разработки программного обеспечения [Электронный ресурс] : учеб. пособие для вузов по направлениям подгот. 09.04.01 и 09.03.03 "Информатика и вычисл. техника" / Л. Г. Гагарина, Е. В. Кокорева, Б. Д. Виснадул ; под ред. Л. Г. Гагариной. - Документ Bookread2. - М. : Форум [и др.], 2017. - 399 с. - Режим доступа: <http://znanium.com/bookread2.php?book=768473>

Списки дополнительной литературы

3. Авдошин, С. М. Программа дисциплины «Введение в программную инженерию» для направления 080700.62 «Бизнес-информатика» подготовки магистра [Электронный ресурс] / С. М. Авдошин ; ГУ ВШЭ. – М. : ГУ ВШЭ, 2008. – Режим доступа: <https://studfiles.net/preview/2428788/>.
4. Албахари, Д. C# 5.0. Справочник. Полное описание языка [Текст] / Д. Албахари, Бен Албахари. = C# 5.0 in a Nutshell: The Definitive Reference. - М. : Вильямс, 2013. - 1008 с.
5. Архипенков, С. Введение в программную инженерию [Электронный ресурс] / С. Архипенков. – Режим доступа : http://citforum.ru/SE/project/arkhipenkov_lectures/.
6. Батоврин, В. К. Системная и программная инженерия [Электронный ресурс] : словарь-справочник : учеб. пособие по направл. «Информ. Системы», «Информатика и вычисл. техника», «Систем. анализ и упр.» / В. К. Батоврин. - М. : ДМК Пресс, 2010. - 280 с. – Режим доступа: <http://ibooks.ru/reading.php?productid=22387>.
7. Боэм, Б. У. Инженерное проектирование программного обеспечения [Текст] / Б. У. Боэм : пер. с англ. ; под ред. А. А. Красиловой. - М. : Радио и связь, 1985. – 512 с.
8. Брауде, Э. Технология разработки программного обеспечения [Текст] / Б. У. Боэм : пер. с англ. - СПб. : Питер, 2004. - 655 с.
9. Вендров, А. М. Проектирование программного обеспечения экономических информационных систем [Текст] : учеб. для вузов по спец. «Прикл. Информатика», «Прикл. Математика и информатика» / А. М. Вендров. - 2-е изд., перераб. и доп. - М. : Финансы и статистика, 2006. - 543 с.
10. Вигерс, К. И. Разработка требований к программному обеспечению [Текст] / К. И. Вигерс : пер. с англ. - М. : Русская редакция, 2004. - 576 с.
11. Гецци, К. Основы инженерии программного обеспечения [Текст] / К. Гецци, М. Джазайери, Д. Мандртоли. - 2-е издание. - СПб. : БХВ-Петербург, 2005. – 832 с.
12. Ехлаков, Ю. П. Введение в программную инженерию [Электронный ресурс] : учеб. пособие / Ю. П. Ехлаков. - Томск : Эль Контент, 2011. – 148 с. – Режим доступа : <http://ibooks.ru/product.php?productid=27931> ; <http://edu.tusur.ru/training/publications/141>.
13. Карпенко, С. Н. Введение в программную инженерию [Электронный ресурс] : учеб.-метод. мат. / С. Н. Карпенко. - Нижний Новгород, 2007. - 103 с. – Режим доступа : <http://www.twirpx.com/file/938473/>.
14. Кознов, Д. Введение в программную инженерию [Электронный ресурс] / Д. Кознов. – Режим доступа : <http://www.intuit.ru/studies/courses/497/353/info>.

15. Константайн, Л. Разработка программного обеспечения [Текст] / Л. Константайн, Л. Локвуд : пер. с англ. В. Шрага. - СПб. : Питер, 2004. - 592 с.
16. Лаврищева, Е. М. Методы и средства инженерии программного обеспечения [Электронный ресурс] : учебник / Е. М. Лаврищева, В. А. Петрухин. - М. : МФТИ (ГУ), 2006. - 304 с. - Режим доступа : <http://window.edu.ru/resource/699/41699>.
17. Лефингвел, Д. Принципы работы с требованиями к программному обеспечению. Унифицированный подход [Текст] / Д. Лефингвел, Д. Уидриг. - М. : Вильямс, 2002. - 448 с.
18. Липаев, В. В. Анализ и сокращение рисков проектов сложных программных средств [Текст] / В. В. Липаев. - М. : СИНТЕГ, 2005. - 208 с.
19. Липаев, В. В. Документирование и управление конфигурацией программных средств. Методы и стандарты [Текст] / В. В. Липаев. - М. : СИНТЕГ, 1998. - 447 с.
20. Липаев, В. В. Документирование сложных программных средств [Текст] / В. В. Липаев. - М. : СИНТЕГ, 2005. - 216 с.
21. Липаев, В. В. Методы обеспечения качества крупномасштабных программных средств [Текст] / В. В. Липаев. - М. : РФФИ СИНТЕГ, 2003. - 520 с.
22. Липаев, В. В. Отладка сложных программ [Текст] / В. В. Липаев. - М. : Энергоатомиздат, 1993. - 382 с.
23. Липаев, В. В. Программная инженерия. Методологические основы [Текст] : учеб. для вузов по направл. «Бизнес-информатика» / В. В. Липаев ; Гос. ун-т Высш. шк. экономики. - М. : ТЕИС, 2006. - 606 с.
24. Липаев, В. В. Проектирование программных средств [Текст] / В. В. Липаев. - М. : Высш. шк., 1990. - 302 с.
25. Липаев, В. В. Системное проектирование сложных программных средств для информационных систем [Текст] / В. В. Липаев. - Изд. 2-е, перераб. и доп. - М. : СИНТЕГ, 2002. - 268 с.
26. Липаев, В. В. Техничко-экономическое обоснование проектов сложных программных средств [Текст] / В. В. Липаев. - М. : СИНТЕГ, 2004. - 284 с.
27. Липаев, В. В. Функциональная безопасность программных средств [Текст] / В. В. Липаев. - М. : СИНТЕГ, 2004. - 348 с.
28. Мацяшек, Л. А. Практическая программная инженерия на основе учебного примера [Текст] / Л. А. Мацяшек, Б. Л. Лионг : пер. с англ. А. М. Епанешникова, В. А. Епанешникова. - М. : БИНОМ. Лаб. знаний, 2009. - 956 с.
29. Мацяшек, Л. Анализ требований и проектирование систем. Разработка информационных систем с использованием UML [Текст] / Л. А. Мацяшек ; пер. с англ. - М. : Вильямс, 2002. - 432 с.
30. Нейгел, К. C# 4: Платформа .NET 4 для профессионалов [Текст] / К. Нейгел и др. = Professional C# 4 and .NET 4. - М. : Диалектика, 2010. - С. 1440.
31. Орлов, С. А. Технологии разработки программного обеспечения. Разработка сложных программных систем [Текст] : учеб. для вузов по спец. «Прогр. обеспечение вычисл. техники и автоматизир. Систем» / С. А. Орлов. - СПб. : Питер, 2004. - 526 с.
32. Орлов, С. А. Технологии разработки программного обеспечения: современный курс по программной инженерии [Текст] : учеб. для вузов по специальности "Прогр. обеспечение вычисл. техники и автоматизир. систем", направления подгот. дипломир. специалистов "Информатика и вычисл. техника" / С. А. Орлов, Б. Я. Цилькер. - 4-е изд. - СПб. : Питер, 2012. - 608 с.
33. Рекомендации по преподаванию программной инженерии и информатики в университетах = Software Engineering 2004: Curriculum Guidelines for Undergraduate Degree Pro-grams in Software Engineering; Computing Curricula 2001: Computer Science [Электронный ресурс] / пер. с англ. - М. : ИНТУИТ.РУ, 2007. - 462 с. - Режим доступа : <http://it.mmcs.sfedu.ru/docs/s2004.pdf>.
34. Смит, Д. C#: программирование для профессионалов [Текст] / Д. Смит. - 2-е издание = C# in Depth, 2nd Edition. - М. : Вильямс, 2011. - 544 с.
35. Соммервилл, И. Инженерия программного обеспечения [Текст] / И. Соммервилл ; пер. с англ. - 6-е изд-е. - М. : Вильямс, 2002. - 624 с.
36. Стилмен, Э. Изучаем C# [Текст] / Э. Стилмен, Дж. Грин. - 2-е издание = Head First C#, 2ed. - СПб. : Питер, 2012. - 704 с.
37. Технология создания программного обеспечения [Электронный ресурс]. - Режим доступа :

http://ndo.sibsubtis.ru/magistr/courses_work/tspo_work/lectures_index.htm.

38. Троелсен, Э. Язык программирования C# 5.0 и платформа .NET 4.5 [Текст] / Э. Троелсен. - 6-е издание = Pro C# 5.0 and the .NET 4.5 Framework, 6th edition. - М. : Вильямс, 2013. - 1312 с.
39. Фатрелл, Р. Т. Управление программными проектами. Достижение оптимального качества при минимуме затрат [Текст] / Р. Т. Фатрелл, Д. Ф. Шафер, Л. И. Шафер. - М. : Вильямс, 2004. - 1140 с.
40. Хейлсберг, А. Язык программирования C#. Классика Computers Science [Текст] / А. Хейлсберг, М. Торгерсен, С. Вилтамут, П. Голд. - 4-е издание = C# Programming Language (Covering C# 4.0), 4th Ed. - СПб. : Питер, 2012. - 784 с.
41. Шилдт, Г. C# 4.0: полное руководство [Текст] / Г. Шилдт. = C# 4.0 The Complete Reference. - М. : Вильямс, 2010. - С. 1056.
42. Якобсон, А. Унифицированный процесс разработки программного обеспечения [Текст] / А. Якобсон, Г. Буч, Дж. Рамбо. - СПб. : Питер, 2002. - 496 с.

8.2. Перечень ресурсов информационно-телекоммуникационной сети "Интернет" (далее - сеть "Интернет"), необходимых для освоения дисциплины

Интернет-ресурсы

1. Software Engineering Conference (Russia) [Электронный ресурс]. – Режим доступа: <http://www.secr.ru/lang/en-en>. - Загл. с экрана.
2. Streamwork [Электронный ресурс]. – Режим доступа: <http://streamwork.ru/upravlenie-proektami/348>. - Загл. с экрана.
3. Введение в программную инженерию и управление жизненным циклом ПО. Общие вопросы управления проектами [Электронный ресурс]. – Режим доступа: <http://bookfi.org/book/523981>. - Загл. с экрана.
4. Введение в программную инженерию [Электронный ресурс]. – Режим доступа: <http://do.gendocs.ru/docs/index-104434.htm>. - Загл. с экрана.
5. Программная инженерия. Управление программной инженерией [Электронный ресурс]. – Режим доступа: http://www.software-testing.ru/files/se/3-7-software_engineering_management.pdf. - Загл. с экрана.
6. Системная инженерия программного обеспечения [Электронный ресурс]. – Режим доступа: <http://www.osr.ru/os/2002/05/181460/>. – Загл. с экрана.
7. Электронная библиотечная система Поволжского государственного университета сервиса [Электронный ресурс]. - Режим доступа: <http://elib.tolgaz.ru/>. - Загл. с экрана.
8. Электронно-библиотечная система Znaniy.com [Электронный ресурс]. - Режим доступа: <http://znaniy.com/>. – Загл. с экрана.
9. Электронно-библиотечная система Лань [Электронный ресурс]. - Режим доступа: <https://e.lanbook.com/books>. - Загл. с экрана.

9. Перечень информационных технологий, используемых при осуществлении образовательного процесса по дисциплине, включая перечень программного обеспечения и информационных справочных систем

Краткая характеристика применяемого программного обеспечения

№ п/п	Программный продукт	Характеристика	Назначение при освоении дисциплины
1	Операционная система Microsoft Windows.	Семейство проприетарных операционных систем (ОС) корпорации Microsoft, ориентированных на применение графического интерфейса при управлении	Предназначен для проведения лабораторных занятий
2	Пакет Microsoft Office (MS Word, MS PowerPoint).	Программное обеспечение для работы с различными типами документов: текстами, электронными таблицами, базами данных и др	Предназначен для проведения лабораторных занятий

3	Visual Studio	Объектно-ориентированный язык программирования	Предназначен для проведения лабораторных занятий
4	Браузер Internet Explorer.	Прикладное программное обеспечение для просмотра веб-страниц, содержания веб-документов, управления веб-приложениями	Предназначен для проведения лабораторных занятий

10. Описание материально-технической базы, необходимой для осуществления образовательного процесса по дисциплине

Для проведения занятий лекционного типа используются специальные помещения - учебные аудитории, укомплектованные специализированной мебелью и техническими средствами обучения, служащими для представления учебной информации.

Для проведения лабораторных работ используется комплексная лаборатория диагностирования и технического обслуживания, оснащенная персональными компьютерами с пакетом Microsoft Office (MS Word, MS Excel, MS PowerPoint, MS Visual Studio Team System (VSTS)) и браузером Internet Explorer.

Для текущего контроля и промежуточной аттестации используются специальные помещения - учебные аудитории, укомплектованные специализированной мебелью, и (или) компьютерные классы, оснащенные компьютерной техникой с возможностью подключения к сети Интернет и обеспечением доступа в электронную информационно-образовательную среду университета.

Для самостоятельной работы обучающихся используются специальные помещения - учебные аудитории для самостоятельной работы, оснащенные компьютерной техникой с возможностью подключения к сети Интернет и обеспечением доступа в электронную информационно-образовательную среду университета.

11. Примерная технологическая карта дисциплины «Введение в программную инженерию»

Факультет информационно-технического сервиса

кафедра «Информационный и электронный сервис»

направление подготовки 09.03.04 «Программная инженерия»

направленности (профиля) «Разработка программно-информационных систем»

4 семестр

№ п/п	Виды контрольных точек	Количество контрольных точек	Количество баллов за 1 контрольную точку	Срок прохождения контрольных точек																					Зачетно-экзаменационная сессия
				январь		февраль					март				апрель				май				июнь		
				1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	
I	Обязательные:																								
1.1	посещение лекционных занятий	9	1				+		+		+		+		+		+		+		+		+		
1.2	защита лабораторных работ	4	4						+		+		+		+										
1.3	подготовка докладов, сообщений на лекционных занятиях	9	2				+		+		+		+		+		+		+		+		+		
1.4	контрольная работа	1	10												+										
1.5	промежуточное тестирование	1	10														+								
1.6	итоговое тестирование	1	10																				+		
II	Творческий рейтинг:																								
2.1	выполнение и защита индивидуальных заданий	1	10																						
2.2	подготовка доклада на научной конференции	1	17															+							
Форма контроля (зачет/экзамен)			100													Атт.									зачет

Факультет информационно-технического сервиса
кафедра «Информационный и электронный сервис»
направление подготовки 09.03.04 «Программная инженерия»
направленности (профиля) «Разработка программно-информационных систем»
5 семестр

№ п/п	Виды контрольных точек	Количество контрольных точек x Количество баллов за 1 контрольную точку	Срок прохождения контрольных точек																	Зачетно- экзамена- ционная сессия	
			Сентябрь				Октябрь					Ноябрь				Декабрь					
			1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17		
I	Обязательные:																				
1.1	посещение лекционных занятий	9	1			+	+	+		+		+		+		+		+		+	
1.2	защита лабораторных работ	9	3				+		+		+		+		+		+	+	+	+	
1.3	подготовка докладов, сообщений на лекционных занятиях	9	2			+	+	+		+		+		+		+		+		+	
1.4	контрольная работа	1	8											+							
1.5	промежуточное тестирование	1	8												+						
1.6	итоговое тестирование	1	10																	+	
II	Творческий рейтинг:																				
2.1	подготовка доклада на научной конференции	1	10														+				
2.2	подготовка рефератов (п. 6.2)	1	10																+		
Форма контроля														Атт.							экзамен

