

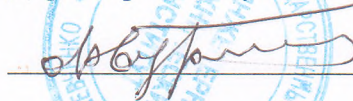
Государственное образовательное учреждение
«Приднестровский государственный университет им. Т.Г. Шевченко»

Инженерно-технический институт

Кафедра «Программное обеспечение вычислительной техники
и автоматизированных систем»

УТВЕРЖДАЮ:

Директор института, доцент



Ф.Ю. Бурменко

«22»

09

2017 г.

РАБОЧАЯ ПРОГРАММА

на 2017/2018 учебный год

УЧЕБНОЙ ДИСЦИПЛИНЫ

Б1.В.ОД.1 «ВВЕДЕНИЕ В ПРОГРАММНУЮ ИНЖЕНЕРИЮ»

Направление подготовки:

09.03.04 Программная инженерия

Профиль подготовки

Разработка программно-информационных систем

Для набора

2016 года

Квалификация (степень) выпускника

бакалавр

Форма обучения:

очная

Тирасполь, 2017

Рабочая программа дисциплины «ВВЕДЕНИЕ В ПРОГРАММНУЮ ИНЖЕНЕРИЮ»
/сост. Е.В. Добровольская – Тирасполь: ГОУ ПГУ, 2017. – 15 с.

Рабочая программа предназначена для преподавания обязательной дисциплины вариативной части дисциплин (модулей) студентам очной формы обучения по направлению подготовки 09.03.04 «ПРОГРАММНАЯ ИНЖЕНЕРИЯ»

Рабочая программа составлена с учетом Федерального Государственного образовательного стандарта высшего образования по направлению подготовки 09.03.04 «ПРОГРАММНАЯ ИНЖЕНЕРИЯ», утвержденного приказом Министерства образования и науки Российской Федерации от 12 марта 2015 г. № 229.

Составитель



/Е.В. Добровольская, ст. преп

«31» августа 2017 г.

1. Цели и задачи освоения дисциплины

Дисциплина «Введение в программную инженерию» закладывает основу для изучения программной инженерии, дает базовые понятия для изучения дисциплин, относящихся к программной инженерии.

Цели дисциплины:

- получение общих знаний в области проектирования, проверки корректности, моделирования жизненного цикла,
- управления качеством программного обеспечения, навыков организации управления проектами по разработке программного обеспечения,
- систематизация представлений о современном комплексе задач, методов и стандартах программной инженерии,
- создание и эволюции сложных, многоверсионных, тиражируемых программных продуктах высокого качества.

Задачи дисциплины: являются введение понятий проектирования, разработки, сопровождения и документирования программных продуктов с использованием регламентированных процессов в соответствии с формальными требованиями, определенными заказчиком.

2. Место дисциплины в структуре ООП ВПО

Б1.В.ОД.1, Дисциплины (модули). Вариативная часть. Обязательные дисциплины.

Общая трудоемкость дисциплины составляет 3 зачетные единицы, 108 часов.

Для освоения дисциплины «Введение в программную инженерию» студенты используют знания, умения, навыки, способы деятельности и установки, полученные и сформированные в ходе изучения дисциплин «Информатика и программирование» и «Введение в профиль», «Программирование», «Объектно-ориентированное программирование», «Базы данных».

Изучение дисциплины «Введение в программную инженерию» является базой для дальнейшего освоения студентами дисциплин «Конструирование программного обеспечения», «Разработка и анализ требований к программному обеспечению», «Управление программными проектами», «Экономика программной инженерии», «Проектирование и архитектура программных систем», «Тестирование программного обеспечения», «Надежность и оптимизация программного обеспечения», курсов по выбору и для прохождения практики.

3. Требования к результатам освоения дисциплины

Изучение дисциплины направлено на формирование следующих компетенций: ОПК-2, ПК-3, ПК-4, ПК-5, ПК-7, ПК-8, ПК-9. Расшифровка компетенций дана в следующей таблице.

Таблица 1 – Формулировка компетенции для направления 09.03.04 «ПРОГРАММНАЯ ИНЖЕНЕРИЯ»

Код компетенции	Формулировка компетенций
ОПК-2	владением архитектурой электронных вычислительных машин и систем
ПК-3	владением навыками использования различных технологий разработки программного обеспечения
ПК-4	владением концепциями и атрибутами качества программного обеспечения (надежности, безопасности, удобства использования), в том числе роли людей,

	процессов, методов, инструментов и технологий обеспечения качества
ПК-5	владением стандартами и моделями жизненного цикла
ПК-7	владением методами управления процессами разработки требований, оценки рисков, приобретения, проектирования, конструирования, тестирования, эволюции и сопровождения
ПК-8	владением основами групповой динамики, психологии и профессионального поведения, специфичных для программной инженерии
ПК-9	владением методами контроля проекта и готовностью осуществлять контроль версий

В результате освоения дисциплины студент должен:

3.1 Знать:

- методологические основы современной программной инженерии, обеспечивающей жизненный цикл сложных программных средств;
- методы и процессы верификации, тестирования и оценивания корректности программных компонентов, а также их интеграции, квалификационного тестирования и испытаний крупных комплексов программ;
- методы и процессы документирования, удостоверения качества и сертификации программных продуктов.

3.2 Уметь

- выполнять проектирование в одной или нескольких предметных областях, используя подходы программной инженерии, объединяющие «этические, социальные, юридические и экономические интересы»;
- работать индивидуально или в группе над созданием качественных программ;
- искать приемлемые компромиссы в рамках ограничений, накладываемых «затратами, временем, знаниями, существующими системами и организацией»;
- применять существующие теории, модели и методы, необходимые для программной инженерии;
- изучать новые модели, методы и технологии по мере их появления.

3.3 Владеть

- навыками проектирования, разработки, создания и сопровождения программного обеспечения;
- современными методами, средствами и технологиями разработки программного обеспечения

Рабочая программа учебной дисциплины рассчитана на 68 часов аудиторных занятий, в том числе 34 часа отводится на лекционные занятия, 34 часа – на лабораторные занятия.

С целью систематизации и закрепления полученных теоретических знаний и практических умений в рабочей программе учебной дисциплины предусмотрена самостоятельная работа студентов. Внеаудиторная самостоятельная работа студентов – 40 часов.

Учебная дисциплина изучается один семестр и заканчивается промежуточной аттестацией студентов в форме зачета.

4. Структура и содержание дисциплины (модуля)

4.1. Распределение трудоемкости в часах по видам аудиторной и самостоятельной работы студента по семестрам

Семестр	Количество часов							Форма итогового контроля
	Трудоемкость з.е./часы	В том числе				Самост. работа	Экзамен	
		Аудиторных						
		Всего	Лекции	Лаб. раб.	Практич. занятия			
4	3/108	68	34	34	-	40	-	зачет
Итого	3/108	68	34	34	-	40	-	зачет

4.2 Распределение видов учебной работы и их трудоемкости по разделам дисциплины:

№ раз дела	Наименование раздела дисциплины (модуля)	Всего	Аудиторная работа			Внеаудиторная работа СР
			ЛК	ПЗ	ЛР	
1	Процесс разработки программного обеспечения (ПО)	8	4	-	-	4
2	Рабочий продукт, дисциплина обязательств, проект	10	4	-	-	6
3	Жизненный цикл программного обеспечения	10	4	-	-	6
4	Управление требованиями и архитектура ПО	22	6	-	10	6
5	Тестирование	20	6	-	8	6
6	Диаграммные техники в работе со знаниями	28	6	-	16	6
7	Модели CMM и CMMI	10	4	-	-	6
Итого:		108	34	-	34	40

4.3 Тематический план по видам учебной деятельности

Лекции

№ п/п	Номер раздела дисциплины	Объем часов	Тема лекции	Учебно- наглядные по- собия
1	1	2	Программная инженерия Программное обеспечение	Слайды презентации
2	1	2	Процесс. Совершенствование процесса. Классические модели процесса	Слайды презентации
3	2	2	Рабочий продукт.	Слайды презентации
4	2	2	Дисциплина обязательств. Проект.	Слайды презентации

№ п/п	Номер раздела дисциплины	Объем часов	Тема лекции	Учебно-наглядные пособия
5	3	2	Понятие жизненного цикла программного обеспечения (ПО). Классическая модель жизненного цикла ПО	Слайды презентации
6	3	2	Итеративная модель жизненного цикла ПО. Спиральная модель жизненного цикла ПО	Слайды презентации
7	4	2	Определение архитектуры ПО. Виды и свойства требований.	Слайды презентации
8	4	2	Варианты формализации требований. Цикл работы с требованиями.	Слайды презентации
9	4	2	Единицы конфигурационного управления. Управление версиями, сборками.	Слайды презентации
10	5	2	Тестирование, его виды	Слайды презентации
11	5	2	Управление качеством.	Слайды презентации
12	5	2	Работа с ошибками.	Слайды презентации
13	6	4	Диаграммные техники в работе со знаниями: Метод «случай использования»	Слайды презентации
14	6	4	Диаграммные техники в работе со знаниями: Метод «случай использования».	Слайды презентации
15	6	2	Итеративный цикл автор/рецензент	Слайды презентации
16	7	2	Модели <i>CMM</i> и <i>CMMI</i> .	Слайды презентации
17	7	2	Уровни зрелости процессов по <i>CMMI</i> .	
	Итого:	34		

Практические (семинарные) занятия

Не предусмотрены

Лабораторные работы

№ п/п	Номер раздела дисциплины	Объем часов	Тема лабораторного занятия	Учебно-наглядные пособия
1	4	2	Разработка требований и постановка задачи проектирования. Разработка технического задания к разрабатываемому продукту	Электронный вариант лаб. раб.
2	4	2	Разработка требований и постановка задачи проектирования. Разработка технического задания к разрабатываемому продукту	Электронный вариант лаб. раб.
3	4	2	Проектирование архитектуры и структуры ПО	Электронный вариант лаб. раб.
4	4	2	Проектирование архитектуры и структуры ПО	Электронный вариант лаб. раб.
5	4	2	Проектирование архитектуры и структуры ПО	Электронный вариант лаб. раб.

№ п/п	Номер раздела дисциплины	Объем часов	Тема лабораторного занятия	Учебно- наглядные пособия
6	5	2	<i>Rational Rose</i> - CASE-средство для автоматизации этапов анализа.	Электронный вариант лаб. раб.
7	5	2	<i>Rational Rose</i> - CASE-средство для автоматизации этапов анализа.	Электронный вариант лаб. раб.
8	5	2	Диаграммы вариантов использования	Электронный вариант лаб. раб.
9	5	2	Диаграммы вариантов использования	Электронный вариант лаб. раб.
10	6	2	Диаграммы взаимодействия (<i>interaction diagrams</i>).	Электронный вариант лаб. раб.
11	6	2	Диаграммы классов.	Электронный вариант лаб. раб.
12	6	2	Диаграммы состояний	Электронный вариант лаб. раб.
13	6	2	Диаграммы состояний	Электронный вариант лаб. раб.
14	6	2	Диаграмма последовательности (<i>sequence diagrams</i>).	Электронный вариант лаб. раб.
15	6	2	Диаграмма последовательности (<i>sequence diagrams</i>).	Электронный вариант лаб. раб.
16	6	2	Диаграмма кооперации (<i>collaboration diagram</i>)	Электронный вариант лаб. раб.
17	6	2	Диаграмма кооперации (<i>collaboration diagram</i>)	Электронный вариант лаб. раб.
Итого:		34		

Самостоятельная работа студента

Раздел дисциплины	№ п/п	Тема и вид СРС	Трудоёмкость (в часах)
Раздел 1	1	Тема: Процесс разработки ПО. СРС №1:- работа студентов с лекционным материалом и раздаточными материалами, - поиск и анализ литературы и электронных источников информации	2
	2	Тема: История возникновения программной инженерии СРС №2:- работа студентов с лекционным материалом и раздаточными материалами, - поиск и анализ литературы и электронных источников информации	2
Раздел 2	3	Тема: Стандарты, применяемые при разработке ПО СРС №3:- работа студентов с лекционным материалом и раздаточными материалами, - поиск и анализ литературы и электронных источников информации	2
Раздел 3	4	Тема: Жизненный цикл ПО СРС №4:- работа студентов с лекционным материалом и раздаточными материалами, - поиск и анализ литературы и электронных источников информации	2

Раздел дисциплины	№ п/п	Тема и вид СРС	Трудоёмкость (в часах)
		формации	
	5	Тема: Модели жизненного цикла ПО СРС №5: - работа студентов с лекционным материалом и раздаточными материалами, - поиск и анализ литературы и электронных источников информации	4
Раздел 4	6	Тема: Разработка и анализ требований СРС №6: - работа студентов с лекционным материалом и раздаточными материалами, - поиск и анализ литературы и электронных источников информации	4
	7	Тема: Виды архитектур программных систем и продуктов СРС №7: - работа студентов с лекционным материалом и раздаточными материалами, - поиск и анализ литературы и электронных источников информации	4
	8	Тема: Конфигурационное управление, системы отслеживания версий, задач, ошибок. СРС №8: - работа студентов с лекционным материалом и раздаточными материалами, - поиск и анализ литературы и электронных источников информации	4
Раздел 5	9	Тема: Виды тестирования, системы разработки тестов СРС №9: - работа студентов с лекционным материалом и раздаточными материалами, - поиск и анализ литературы и электронных источников информации	4
Раздел 6	10	Тема: Диаграммы, используемые для проектирования и разработки архитектуры ПО СРС №10: - работа студентов с лекционным материалом и раздаточными материалами, - поиск и анализ литературы и электронных источников информации	4
Раздел 7	11	Тема: История и текущий статус <i>MSF</i> . Модели командной разработки ПО СРС №11: - работа студентов с лекционным материалом и раздаточными материалами, - поиск и анализ литературы и электронных источников информации	4
Раздел 8	12	Тема: Модели зрелости ПО СРС №12: - работа студентов с лекционным материалом и раздаточными материалами, - поиск и анализ литературы и электронных источников информации	4
		Итого:	40

5. Примерная тематика курсовых проектов (работ)

Курсовые работы не предусмотрены

6. Образовательные технологии

Семестр	Вид занятия (Л, ПР, ЛР)	Используемые интерактивные образовательные технологии	Количество часов
4	Л	Компьютерные технологии обучения, проблемная лекция, лекция-диспут, лекция-визуализация: информационно-развивающие технологии, <i>case-study</i> (ситуационный анализ)	34
	ЛР	Компьютерные технологии обучения, Технологии учебного проектирования и моделирования; Ситуационный анализ	34
Итого:			68

7. Оценочные средства для текущего контроля успеваемости, промежуточной аттестации по итогам освоения дисциплины и учебно-методическое обеспечение самостоятельной работы студентов

Предполагается выполнение с последующей защитой лабораторных работ, модульный контроль для оценки уровня усвоения материала студентами

Примерный вариант модульного контроля №1:

1. Промышленные программные продукты – это

- а) ☐ большие программные системы, которые применяются для решения сложных задач, имеют большое время жизни и от их функционирования зависит большое количество пользователей
- б) ☐ программы, используемые в промышленности

2. Артефакт, произведенный в процессе разработки ПО, например, файл или набор файлов, документы, составные части продукта, сервисы, процессы, спецификации, счета – это

- а) ☐ обязательства б) ☐ рабочий продукт в) ☐ проект

3. Укажите виды первичных программных ошибок

- а) ☐ программные ошибки из-за неправильной записи текста программ на языке программирования и ошибок трансляции программы в объектный код
- б) ☐ системные ошибки, обусловленные отклонением функционирования ПС в реальной системе и отклонениями характеристик внешних объектов от предполагаемых при проектировании
- в) ☐ технологические ошибки подготовки машинных носителей и документации, а также ввода программ в память ПК
- г) ☐ сбои, которые не отражаются на работоспособности программ, ущербом от которых можно пренебречь
- д) ☐ алгоритмические ошибки, связанные с неполным формулированием необходимых условий решения и некорректной постановкой задач
- е) ☐ катастрофические отказы, ущерб от которых отражает безопасность данного ПС
- ж) ☐ ординарные отказы, ущерб от которых находится в некоторых допустимых пределах

4. Отказ – это

- а) ☐ состояние, в котором программа не способна выполнять требуемые функции

- b) ☐ программная аномалия, некорректное определение операций, процесса, данных в программе или программной спецификации
- c) ☐ событие, которое заключается в прекращении программой выполнять требуемые функции с учетом заданных ограничений

5. Стратегия к внедрению инноваций, в которой применяется широкомасштабное внедрение инноваций из стратегических соображений

- a) ☐ *organization pull* b) ☐ *technology push*

6. Определенный тип работы, выполняемый в процессе разработки ПО

- a) ☐ модель процесса b) ☐ фаза c) ☐ вид деятельности

7. Модель, в которой впервые были выделены шаги разработки, четко определенные и следующие в точном порядке

- a) ☐ спиральная b) ☐ водопадная (каскадная) c) ☐ инкрементная

8. Для разработки небольших проектов или при разработке типовых систем обычно используется следующая модель

- a) ☐ спиральная b) ☐ водопадная (каскадная) c) ☐ инкрементная

9. Внутренняя структура продукта (компоненты и их связи), основы пользовательского интерфейса продукта, представление извне

- a) ☐ структура ПО b) ☐ архитектура ПО

10. Виды деятельности при работе с требованиями

- a) ☐ валидация требований b) ☐ сертификация требований
c) ☐ описание требований d) ☐ установка требований
e) ☐ анализ требований f) ☐ выделение требований

11. Язык UML

- a) ☐ может поддерживать любой объектно-ориентированный язык программирования
b) ☐ зависит от языка программирования

12. Диаграммы компонентов предназначены для

- a) ☐ представления общей конфигурации и топологии распределенных программных систем и показа распределения компонентов по отдельным узлам системы
b) ☐ установления зависимости между программными компонентами

13. Виды синхронизации на диаграмме деятельности

- a) ☐ дорожки b) ☐ слияние c) ☐ развилки d) ☐ действия

14. Вид связи на диаграмме классов, при которой объекты класса А включают в себя объект класса В, причем последние не могут разделяться и существуют только в рамках объемлющих объектов

- a) ☐ ассоциация b) ☐ композиция c) ☐ агрегация
d) ☐ ссылочная связь e) ☐ связь по наследованию

15. Какие виды связей используются между актерами в диаграмме вариантов использования:

- a) ☐ включение b) ☐ расширение c) ☐ обобщение

Задание 2. Вписать ответы

1. Перечислите статические диаграммы *UML*, используемые для проектирования архитектуры ПО

- a) _____ b) _____
c) _____ d) _____

2. Перечислить последовательность этапов разработки ПО в классическом жизненном цикле ПО (водопадная модель)

- a) _____ b) _____
c) _____ d) _____
e) _____ f) _____

3. Виды деятельности при работе с требованиями

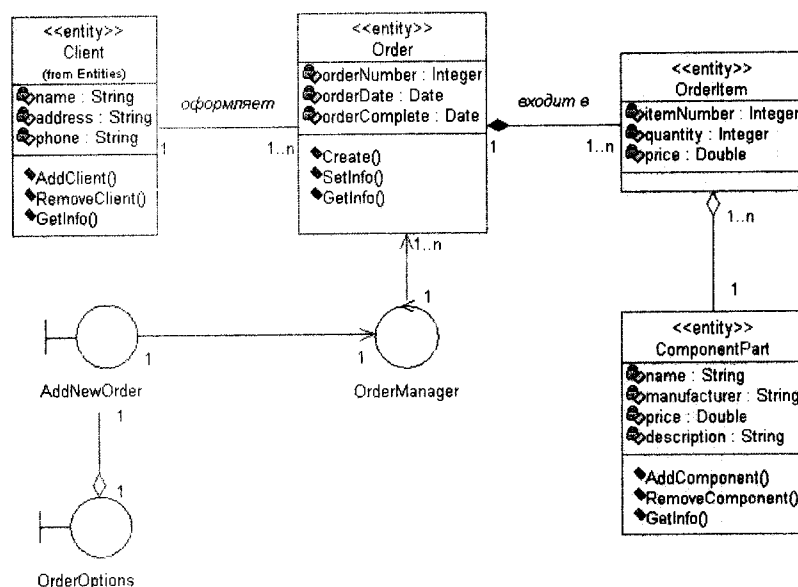
- a) _____
b) _____
c) _____
d) _____

4. В чем преимущества и недостатки использования при разработке ПО водопадной модели и спиральной модели жизненного цикла ПО

5. Достоинства, недостатки, возможные риски при использовании стратегий *organization pull* и *technology push*, привести пример для каждой из стратегий к внедрению инноваций

Задание 3.

Описать диаграмму, представленную на рисунке, опишите ее (объекты, виды связей, назначение, на каком этапе жизненного цикла ПО применяется)



Вид _____

Назначение _____

Этап жизненного цикла ПО _____

Примерный вариант модульного контроля №2:

Задание 1.

1. Укажите виды первичных программных ошибок

- a) ☐ программные ошибки из-за неправильной записи текста программ на языке программирования и ошибок трансляции программы в объектный код
- b) ☐ системные ошибки, обусловленные отклонением функционирования ПС в реальной системе и отклонениями характеристик внешних объектов от предполагаемых при проектировании
- c) ☐ технологические ошибки подготовки машинных носителей и документации, а также ввода программ в память ПК
- d) ☐ сбои, которые не отражаются на работоспособности программ, ущербом от которых можно пренебречь
- e) ☐ алгоритмические ошибки, связанные с неполным формулированием необходимых условий решения и некорректной постановкой задач
- f) ☐ катастрофические отказы, ущерб от которых отражает безопасность данного ПС
- g) ☐ ординарные отказы, ущерб от которых находится в некоторых допустимых пределах

2. Выберите динамические диаграммы UML, используемые для проектирования архитектуры ПО

- | | |
|---|--|
| a) ☐ диаграммы вариантов использования | b) ☐ диаграммы классов |
| c) ☐ диаграммы объектов | d) ☐ диаграммы компонентов |
| e) ☐ диаграммы развертывания | f) ☐ диаграммы деятельности |
| g) ☐ диаграммы сценариев | h) ☐ диаграммы взаимодействия |
| j) ☐ диаграммы состояний | |

3. Диаграммы компонентов предназначены для

- a) ☐ представления общей конфигурации и топологии распределенных программных систем и показа распределения компонентов по отдельным узлам системы
- b) ☐ установления зависимости между программными компонентами

4. Виды синхронизации на диаграмме деятельности

- | | | | |
|-------------------------------------|-------------------------------------|--------------------------------------|--------------------------------------|
| a) ☐ дорожки | b) ☐ слияние | c) ☐ развилки | d) ☐ действия |
|-------------------------------------|-------------------------------------|--------------------------------------|--------------------------------------|

5. Диаграммы, показывающие одну и ту же информацию, но одна привязана к обмену сообщениями во времени, а другая к связи между компонентами

- | | |
|---|--|
| a) ☐ диаграммы вариантов использования | b) ☐ диаграммы классов |
| c) ☐ диаграммы объектов | d) ☐ диаграммы компонентов |
| e) ☐ диаграммы развертывания | f) ☐ диаграммы деятельности |
| g) ☐ диаграммы сценариев | h) ☐ диаграммы взаимодействия |
| j) ☐ диаграммы состояний | |

6. Тестирование – это

- a) ☐ нахождение ошибок и внесение исправлений
- b) ☐ доказательство того, что программа не содержит ошибок
- c) ☐ выполнение программы с целью обнаружения ошибок

7. Вид тестирования, при котором реализация системы недоступна тестирующим, тестируется интерфейс

- | | |
|--|---|
| a) ☐ структурное тестирование | b) ☐ функциональное тестирование |
|--|---|

8. CMMI – это

- a) ☐ модель разработки ПО b) ☐ описанием идеального процесса разработки ПО
c) ☐ модель оценки качества ПО

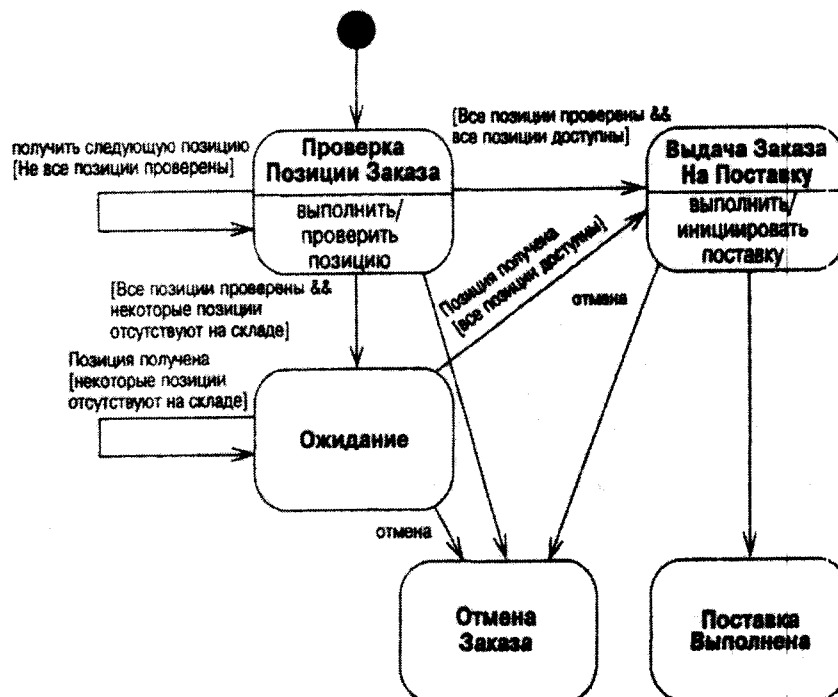
9. Выберите верные утверждения:

- a) ☐ постановка задачи - один из этапов ЖЦПО
b) ☐ необходимо документировать ход выполнения работы
c) ☐ спиральная модель ЖЦПО допускает корректировки во время разработки

10. Оформление постановки задачи по стандартным правилам:

- a) ☐ является одним из этапов жизненного цикла программы
b) ☐ является ресурсоемкой операцией
c) ☐ уменьшает вероятность разночтения пунктов задания разработчиками и заказчиком

Задание 2. Какая диаграмма представлена на рисунке, опишите ее



8. Учебно-методическое и информационное обеспечение дисциплины (модуля):

- 1) Кознов Д.В. Ведение в программную инженерию: 2 издание, исправленное. Национальный открытый университет «ИНТУИТ», 2016
- 2) Мацяшек Л. Практическая программная инженерия на основе учебного примера: учеб. пособие для вузов. - М.:БИНОМ, 2010.– 956 с.
- 3) Орлов С.А. Технологии разработки программного обеспечения: Учебник / – 4-е изд. – СПб. Питер, 2012.– 608 с.

8.2. Дополнительная литература

- 4) В.В. Липаев. Программная инженерия. Методологические основы. Учебник. М.: ТЕИС, 2006.

5) К. Геци, М. Джазайери, Д. Мандртоли. Основы инженерии программного обеспечения, 2-е издание, СПб.: БХВ-Петербург, 2005.

6) Иан Коммервилл. Инженерия программного обеспечения, 6-е издание, М.: Издательский дом "Вильямс", 2002.

8.3. Программное обеспечение и Интернет-ресурсы

Программное обеспечение: ОС *Windows*, *Rational Rose 2000*, *UML*

Интернет-ресурсы

1) Software Engineering Conference (Russia) 2005, 2006, 2007 <http://www.secr.ru/>

2) Software Engineering – Guide to the Software Engineering Body of Knowledge (SWE-BOK) TECHNICAL REPORT ISO/IEC TR 19759 IEEE First edition 2005-09-15. <http://www.secr.ru/>

3) CMMI® for Development, Version 1.2, CMU/SEI-2006-TR-008 ESC-TR-2006-008

8.4. Методические указания и материалы по видам занятий

Методические указания к лабораторным работам по дисциплине «Введение в программную инженерию» в электронном варианте.

9. Материально-техническое обеспечение дисциплины

Учебный кабинет, лаборатория ИТО ИТИ.

10. Методические рекомендации по организации изучения дисциплины:

Обучающийся, изучающий дисциплину, должен, с одной стороны, овладеть общим понятийным аппаратом, а с другой стороны, должен научиться применять теоретические знания на практике.

В результате изучения дисциплины обучающийся должен знать основные определения, понятия, аксиомы, методы доказательств.

Успешное освоение курса требует напряженной самостоятельной работы обучающегося. В программе курса отведено минимально необходимое время для работы обучающегося над темой. Самостоятельная работа включает в себя:

- чтение и конспектирование рекомендованной литературы;
- проработку учебного материала (по конспектам занятий, учебной и научной литературе), подготовку ответов на вопросы, предназначенные для самостоятельного изучения, доказательство отдельных утверждений, свойств, решение задач;
- подготовка к зачету.

Руководство и контроль за самостоятельной работой обучающегося осуществляется в форме индивидуальных консультаций.

Важно добиться понимания изучаемого материала, а не механического его запоминания. При затруднении изучения отдельных тем, вопросов следует обращаться за консультациями к лектору.

Рабочая учебная программа по дисциплине «Введение в программную инженерию» составлена в соответствии с требованиями Федерального Государственного образовательного стандарта ВО и учебного плана по профилю подготовки «Разработка программно-информационных систем».

ТЕХНОЛОГИЧЕСКАЯ КАРТА ДИСЦИПЛИНЫ

Курс 2

Семестр 4

Группа ИТ16ДР62ПИ1

Преподаватель – лектор Добровольская Е.В.

Преподаватели, ведущие лабораторные занятия – Добровольская Е.В.

Кафедра программного обеспечения вычислительной техники и автоматизированных систем

Наименование дисциплины/курса	Уровень образования (бакалавриат, специалитет, магистратура)	Статус дисциплины в учебном плане (А, Б, В)	Количество зачетных единиц	
Введение в программную инженерию	бакалавриат		3	
СМЕЖНЫЕ ДИСЦИПЛИНЫ ПО УЧЕБНОМУ ПЛАНУ:				
«Конструирование программного обеспечения», «Разработка и анализ требований к программному обеспечению», «Управление программными проектами», «Экономика программной инженерии», «Проектирование и архитектура программных систем», «Тестирование программного обеспечения», «Надежность и оптимизация программного обеспечения»				
БАЗОВЫЙ МОДУЛЬ (проверка знаний и умений по дисциплине)				
Тема, задание или мероприятие текущего контроля	Виды текущей аттестации	Аудиторная или внеаудиторная	Минимальное количество баллов	Максимальное количество баллов
Посещение занятий			5	10
Тест 1	Т1	Аудиторная	12	25
Лабораторная работа №1	ЛР1	Аудиторная	3	6
Лабораторная работа №2	ЛР2	Аудиторная	3	7
Лабораторная работа №3	ЛР3	Аудиторная	4	7
РУБЕЖНЫЙ КОНТРОЛЬ	РК1		22	45
Тест 2	Т2	Аудиторная	13	25
Лабораторная работа №4	ЛР3	Аудиторная	3	6
Лабораторная работа №5	ЛР4	Аудиторная	3	7
Лабораторная работа №6	ЛР2	Аудиторная	4	7
РУБЕЖНАЯ АТТЕСТАЦИЯ	РА		23	45
Итого			50	100

Составитель, ст.преп.



Е.В. Добровольская

Рабочая учебная программа рассмотрена методической комиссией инженерно-технического института протокол № 1 от «22» 09 20 17 г. и признана соответствующей требованиям Федерального Государственного образовательного стандарта и учебного плана по направлению 09.03.04 «ПРОГРАММНАЯ ИНЖЕНЕРИЯ».

Председатель МК ИТИ



Е.И. Андрианова

Зав. обслуживающей кафедры, доцент



С.Г. Федорченко