

Министерство образования и науки РФ

Федеральное государственное бюджетное образовательное учреждение  
высшего образования  
«Сибирский государственный индустриальный университет»

Кафедра прикладных информационных технологий и программирования

УТВЕРЖДАЮ

Проректор по учебной работе –  
первый проректор

\_\_\_\_\_ Феоктистов А.В.

« \_\_\_\_ » \_\_\_\_\_ 20 \_\_\_\_ г.

**ПРОГРАММА  
УЧЕБНОЙ ДИСЦИПЛИНЫ**

**Программная инженерия**

09.03.03 – Прикладная информатика

Прикладная информатика в информационной сфере

Квалификация выпускника  
бакалавр

Форма обучения  
заочная

Новокузнецк  
2018

## 1 Цели и задачи освоения учебной дисциплины

В настоящее время происходит интенсивное развитие подходов и парадигм программирования, возникают и развиваются языки программирования. Поэтому профессионалы в области информационных технологий должны обладать навыками свободного обращения с компьютером, владения рядом прикладных программ и операционных систем, программирования на языках программирования высокого уровня. А также профессионалы в области информационных технологий должны знать принципы и особенности современных подходов к программированию, одним из самых актуальных подходов является объектно-ориентированный. Знание технологии проектирования, разработки, тестирования и эксплуатации программного продукта и умение применить свои знания на практике способствует становлению зрелого мышления профессионала по информационным технологиям и хорошему знанию методов современного программирования, что положительным образом скажется на его дальнейшем профессиональном самоопределении.

Основной **целью** курса является формирование у обучающихся общей культуры использования современных подходов к программированию, основываясь на современных требованиях информационных технологий. Понимание основных принципов, технологий построения программного обеспечения информационных систем общего назначения позволит выпускнику лучше ориентироваться и должным образом проявить себя на современном рынке информационных технологий.

**Задачами** курса являются:

- систематическое изложение лекционного материала по принципам применения современных процессов проектирования, разработки, тестирования и эксплуатации программного продукта и о взаимосвязи всех аспектов программной инженерии;
- изучение и сравнительный анализ современных процессов проектирования и разработки программных продуктов;
- изучение принципов и методов оценки качества и управления качеством программного продукта;
- формирование практических навыков по созданию программ на основе объектно-ориентированного подхода с использованием различных языков программирования;
- приобретение практических навыков формирования и анализа требований, оценки качества и тестирования программных продуктов.

## 2 Место учебной дисциплины в структуре ООП по направлению подготовки

Дисциплина «Программная инженерия» входит в вариативную часть (раздел «Обязательные дисциплины») учебного плана бакалавров и является обязательной для изучения.

Для успешного освоения дисциплины обучающиеся должны иметь практические навыки создания программ с использованием языков программирования высокого уровня.

Для успешного изучения дисциплины обучающиеся используют знания и умения, сформированные в ходе изучения дисциплин «Содержательные основы прикладной информатики и вычислительной техники», «Основы программирования», «Программирование».

Изучение данной дисциплины является основой для последующего изучения дисциплин «Инфокоммуникационные системы и сети», «Информационная безопасность и защита информации», «Web-технологии», «Web-программирование», «Проектирование информационных систем», а также для прохождения практик, выполнения курсовых проектов и работ, последующей подготовки к государственной итоговой аттестации.

### **3 Планируемые результаты обучения по учебной дисциплине**

Процесс изучения учебной дисциплины «Программная инженерия» направлен на формирование следующих компетенций:

#### **– профессиональные компетенции (ПК):**

*ПК-1* - способностью проводить обследование организаций, выявлять информационные потребности пользователей, формировать требования к информационной системе.

Структура компетенции:

- знать: экономико-правовые основы разработки программных продуктов;

- уметь: формулировать требования к создаваемым программным комплексам;

- владеть: навыками разработки программных комплексов для решения прикладных задач, оценки сложности алгоритмов и программ, использования современных технологий программирования, тестирования и документирования программных комплексов.

*ПК-2* - способностью разрабатывать, внедрять и адаптировать прикладное программное обеспечение.

Структура компетенции:

- знать: задачи и методы исследования и обеспечения качества и надежности программных компонентов и систем;

- уметь: использовать международные и отечественные стандарты;

- владеть: навыками разработки программных комплексов для решения прикладных задач, оценки сложности алгоритмов и программ, использования современных технологий программирования, тестирования и документирования программных комплексов.

*ПК-8* - способностью программировать приложения и создавать программные прототипы решения прикладных задач.

Структура компетенции:

- знать: принципы организации проектирования, разработки и содержания этапов процесса разработки программных комплексов;
- уметь: применять современные подходы и технологии программирования;
- владеть: навыками разработки программных комплексов для решения прикладных задач, оценки сложности алгоритмов и программ, использования современных технологий программирования, тестирования и документирования программных комплексов.

**– профессионально-специализированные компетенции (ПСК):**

*ПСК-1* - способностью проектировать программные комплексы, базы данных, автоматизированные информационные системы на основе современных инструментальных средств и технологий программирования.

Структура компетенции:

- знать: профили открытых информационных систем, функциональные и технологические стандарты разработки программных комплексов;
- уметь: формировать архитектуру программных комплексов для информатизации предприятий, разрабатывать программные приложения;
- владеть: навыками разработки программных комплексов для решения прикладных задач, оценки сложности алгоритмов и программ, использования современных технологий программирования, тестирования и документирования программных комплексов.

#### **4 Структура и содержание учебной дисциплины**

Программой учебной дисциплины «Программная инженерия» предусмотрено проведение лекций и практических занятий, руководство курсовой работой. Особое место в овладении учебной дисциплины «Программная инженерия» отводится самостоятельной работе, позволяющей получить максимальное представление о данной учебной дисциплине.

Контактная работа обучающихся с преподавателем включает в себя занятия лекционного типа (лекции) практические занятия, групповые консультации, индивидуальную работу обучающихся с преподавателем, промежуточную аттестацию обучающихся и иную контактную работу, предусматривающую групповую или индивидуальную работу обучающихся с преподавателем. Контактная работа обучающихся с преподавателем может быть аудиторной, внеаудиторной, а также проводиться в электронной информационно-образовательной среде.

Дисциплина проводится в течение третьего курса.

Общая трудоемкость дисциплины «Программная инженерия» (3 курс) составляет 5 зачетных единиц (180 академических часов), из них

24 аудиторных часа и 156 часов на самостоятельную работу (включая 9 часов на подготовку к экзамену).

### Тематический план учебной дисциплины «Программная инженерия»

Наименование разделов и тем учебной дисциплины	Количество часов				
	всего	в том числе			самостоятельная работа
		аудиторные			
		лекции	ЛР	ПЗ	
<b>Раздел 1. Технология разработки ПС</b>					
1.1 Определение программной инженерии	5	1			4
1.2 Методы проектирования архитектуры системы	11	1		2	8
1.3 Методы спецификаций ПО	10			2	8
1.4 Интерфейсы	13	1			12
Выполнение курсовой работы	10				10
<i>Итого по разделу 1</i>	<b>49</b>	<b>3</b>		<b>4</b>	<b>42</b>
<b>Раздел 2. Управление требованиями и качеством</b>					
2.1 Методы определения требований	11	1			10
2.2 Методы анализа требований	10			2	8
2.3 Характеристика качества ПО	11	1		2	8
2.4 Метрики качества	12	1		2	9
2.5 Сложность ПС	6				6
2.6 Корректность ПС	6				6
2.7 Надежность ПС	6				6
Выполнение курсовой работы	10				10
<i>Итого по разделу 2</i>	<b>72</b>	<b>3</b>		<b>6</b>	<b>63</b>
<b>Раздел 3. Тестирование программного продукта</b>					
3.1 Виды и методы тестирования	7	1			6
3.2 Эвристические методы создания тестов	8			2	6
3.3 Документирование тестирования	9	1		2	6
3.4 Автоматизация тестирования	16			2	14
Выполнение курсовой работы	10				10
<i>Итого по разделу 3</i>	<b>50</b>	<b>2</b>		<b>6</b>	<b>42</b>
Экзамен	9				9
<b>Всего по дисциплине (часов)</b>	<b>180</b>	<b>8</b>		<b>16</b>	<b>156</b>
<b>Всего по дисциплине (зачетных единиц)</b>	<b>5</b>				
Вид промежуточной аттестации (зачет, экзамен)	экзамен, дифференциальный зачет по курсовой работе				

# Содержание учебной дисциплины «Программная инженерия»

## Раздел 1. Технология разработки программного средства

### *Тема 1.1. Определение программной инженерии*

Определение программной инженерии, ее место в инженерной деятельности при создании компьютерных систем и общее описание десяти областей знания профессионального ядра знаний SWEBOOK. Жизненный цикл стандарта ISO/IEC 12207 и связь его процессов с областями знаний SWEBOOK.

### *Тема 1.2. Методы проектирования архитектуры системы*

Методы объектного анализа и построения моделей предметных областей. Объектно-ориентированные и стандартизованные методы проектирования архитектуры системы.

### *Тема 1.3. Методы спецификации*

Принципы разработки спецификации ПО, шаблон спецификации ПО, анализ языков формальной спецификации программ, методы доказательства правильности программ, валидация и верификация программ.

### *Тема 1.4. Интерфейсы*

Программные интерфейсы, взаимодействие разноразличных программ, реинженерия систем, рефакторинг компонентов, основные принципы проектирования пользовательских интерфейсов ПО, юзабилити и тестирование юзабилити ПО.

## Раздел 2. Управление требованиями и качеством

### *Тема 2.1. Методы определения требований.*

Методы определения требований в программной инженерии: сбор, накопление, спецификации и классификация требований.

### *Тема 2.2. Методы анализа требований.*

Методы анализа требований. Структурный анализ: диаграммы потоков данных, описание потоков данных и процессов. Методы анализа, ориентированные на структуры данных. Метод анализа Джексона.

### *Тема 2.3. Характеристики качества программного обеспечения.*

Внутренние и внешние характеристика качества ПО. Методики повышения качества ПО и оценка их эффективности. Стандарты IEEE, связанные с качеством ПО. Закон контроля качества ПО, CMM (модель зрелости процесса разработки ПО).

### *Тема 2.4. Метрики качества.*

Метрики качества. Метрики объектно-ориентированных программных систем (специфика). Набор метрик Чидамбера-Кемерера. Метрики Лоренца и Кидда. Метрики Абреу.

#### *Тема 2.5. Сложность программных средств.*

Основные виды сложности проектирования и функционирования ПС. Показатели вычислительной сложности: временная, программная, информационная сложность и основные факторы, влияющие на их значение. Измерение и оценка сложности программных средств.

#### *Тема 2.6. Корректность программных средств.*

Основные понятия и виды корректности программ. Функциональная, детерминированная, стохастическая динамическая корректность. Типы эталонов, методы измерений и проверки корректности программ. Ошибки в ПС. Количественное описание ошибок ПС. Классификационная схема программных ошибок. Источники ошибок. Применение метрики ПС для обнаружения и устранения ошибок.

#### *Тема 2.7. Надежность программных средств.*

Определение надежности ПС. Показатели надежности ПС. Факторы, определяющие надежность ПС. Определение показателей надежности на различных этапах жизненного цикла ПС. Аналитические, имитационные, экспериментальные методы оценки надежности ПС. Моделирование и обеспечение надежности в процессе создания ПС. Статические, динамические, эмпирические модели.

### **Раздел 3. Тестирование программного продукта**

#### *Тема 3.1. Виды и методы тестирования.*

Терминология: тестирование, отладка, дефект, отказ, сбой. Объекты тестирования. Роль тестирования в различных процессах разработки ПО. Уровни и виды тестирования: модульное, интеграционное, системное. Регрессионное тестирование, smoketesting. Тестирование белого и черного ящика. Виды дефектов, обнаруживаемые на каждом уровне. Нисходящее и восходящее тестирование. Категории тестов системного тестирования: полнота решения функциональных задач; тестирование целостности; стрессовое тестирование; корректность использования ресурсов; оценка производительности; эффективность защиты от искажения данных и некорректных действий; проверка инсталляции и конфигурации на разных платформах; корректность документации. Проблемы регрессионного тестирования. Приемочное тестирование.

#### *Тема 3.2. Эвристические методы создания тестов.*

Эвристические методы создания тестов. Характеристики хорошего теста. Классы эквивалентности исходных данных. Тестирование переходов между состояниями. Тестирование гонок. Нагрузочные тесты. Тестирование юзабилити.

### *Тема 3.3. Документирование тестирования.*

Жизненный цикл дефекта. Версии программного продукта, системы контроля версий. Версии программного продукта и их связь с количеством дефектов. Точка конвергенции. Количественные критерии качества тестирования. Системы документирования дефектов. Категории классификации дефектов: серьезность, приоритет. Принципы описания дефекта. Набор документов для тестирования: функциональная спецификация, спецификация программных требований, матрица прослеживаемости, тест-план, тестовая спецификация, журнал. Состав, назначение и принципы организации тест-плана. Разработка тестового плана. Компоненты тест-плана. Тестовая спецификация: структура, оптимизация, разработка. Тест-лог (журнал) и его анализ. Тестовые примеры (тест-кейсы): структура, принципы разработки. Тестирование белого ящика: классы критериев (структурные, функциональные, стохастические, мутационные), проблемы. Методы создания тестов на основе управляющего графа программы.

### *Тема 3.4. Автоматизация тестирования.*

Автоматизация тестирования: область применения, виды, инструменты, проблемы.

## **5 Перечень тем практических занятий**

№ раздела	Тема практического занятия	Трудоемкость, час.
<b>1</b>	<b><i>Технология разработки ПС</i></b>	<b>4</b>
1.2	Объектно-ориентированные методы проектирования	2
1.3	Методы доказательства и верификации программ	2
<b>2</b>	<b><i>Управление требованиями и качеством</i></b>	<b>6</b>
2.2	Исследование методов анализа требований	2
2.3	Определение характеристик качества программного обеспечения	2
2.4	Исследование метрик объектно-ориентированных программных систем	2
<b>3</b>	<b><i>Тестирование программного продукта</i></b>	<b>6</b>
3.2	Разработка эвристических тестов и тестирование приложений	2
3.3	Создание тестов на основе управляющего графа программы	2
3.4	Создание и использование Unit-тестов	2
<b>Всего</b>		<b>16</b>



## 6 Перечень тем курсовых проектов, работ (расчетно-графических работ, рефератов и др.)

Курсовая работа, трудоемкостью 30 часов, выполняется в течение 3 курса и охватывает все разделы учебной дисциплины.

Курсовая работа заключается в создании собственной программы с помощью современных средств программирования, анализа функциональности, надежности и тестирования разработанной программы и полного документирования всех этапов ее разработки. В таблице приведены возможные темы курсовой работы.

№ раздела дисциплины	Наименование КР	Трудоемкость (час.)
1-3	Система документооборота предприятия	30
1-3	Система учета (материалов, энергоносителей, кадров на предприятии)	30
1-3	Информационно-справочная система в какой-либо области	30
1-3	Информационная система предприятия	30
1-3	Система поддержки принятия решений на предприятии	30

## 7 Виды самостоятельной работы

На самостоятельную работу обучающихся отводится 156 часов, в том числе на подготовку к лекциям, практическим занятиям, выполнение курсовой работы, подготовку к экзамену.

№ раздела/ темы дисциплины	Вид самостоятельной работы	Трудоемкость (час.)
1	1 Изучение лекционного материала. 2 Подготовка к практическим занятиям. 3 Подготовка к текущему контролю. 4 Выполнение курсовой работы.	42
2	1 Изучение лекционного материала. 2 Подготовка к практическим занятиям. 3 Подготовка к текущему контролю. 4 Выполнение курсовой работы.	63
3	1 Изучение лекционного материала. 2 Подготовка к практическим занятиям. 3 Подготовка к текущему контролю. 4 Выполнение курсовой работы.	42
<i>Экзамен</i>	<i>Подготовка к экзамену.</i>	<i>9</i>
<b>Итого</b>		<b>156</b>

## 8 Учебно-методическое и информационное обеспечение учебной дисциплины

### **а) основная литература**

1 Деменков М. Е. Современные методы и средства проектирования информационных систем [Электронный ресурс] : учебное пособие / М. Е. Деменков, Е. А. Деменкова. – Электрон. дан. – Архангельск : ИД САФУ, 2015. – Режим доступа:

<http://www.studentlibrary.ru/book/ISBN9785261011149.html>.

2 Схиртладзе А. Г. Проектирование единого информационного пространства виртуальных предприятий [Электронный ресурс] : учебник / А. Г. Схиртладзе, А. В. Скворцов, Д. А. Чмырь – Электрон. дан. – Москва : Абрис, 2012. – Режим доступа:

<http://www.studentlibrary.ru/book/ISBN9785437200742.html>.

3 Подбельский В. В. Язык Си#. Решение задач [Электронный ресурс] : учебное пособие / В. В. Подбельский. – Электрон. дан. – Москва : Финансы и статистика, 2014. – Режим доступа:

<http://www.studentlibrary.ru/book/ISBN9785279035533.html>.

### **б) дополнительная литература**

1 Мякишев Д. В. Принципы и методы создания надежного программного обеспечения АСУТП [Электронный ресурс] : методическое пособие / Д. В. Мякишев. – Электрон. дан. – Москва : Инфра-Инженерия, 2017. – Режим доступа:

<http://www.studentlibrary.ru/book/ISBN9785972901791.html>.

2 Хеффельфингер Д. Разработка приложений Java EE 6 в NetBeans 7 [Электронный ресурс] / Д. Хеффельфингер ; Пер. с англ.: Е.Н. Карышев. – Электрон. дан. - Москва : ДМК Пресс, 2013. – Режим доступа: <http://www.studentlibrary.ru/book/ISBN9785940749141.html>.

3 Антамошкин О. А. Программная инженерия. Теория и практика [Электронный ресурс] / О. А. Антамошкин. – Электрон. дан. – Красноярск : СФУ, 2012. – Режим доступа:

<http://www.studentlibrary.ru/book/ISBN9785763825114.html>.

4 Нечаев Д. Ю. Надежность информационных систем [Электронный ресурс] / Д. Ю. Нечаев, Ю. В. Чекмарев. – Электрон. дан. – Москва : ДМК Пресс, 2012. – Режим доступа:

<http://www.studentlibrary.ru/book/ISBN9785940745662.html>.

5 Мацяшек Л. А. Практическая программная инженерия на основе учебного примера [Электронный ресурс] / Л. А. Мацяшек. – Электрон. дан. – Москва : БИНОМ, 2012. – Режим доступа:

<http://www.studentlibrary.ru/book/ISBN9785996311828.html> (.).

### **в) ресурсы информационно-телекоммуникационной сети «Интернет»:**

1 Электронный каталог Научно-технической библиотеки СибГИУ [Электронный ресурс] : база данных содержит сведения о всех видах литературы, поступающей в фонд НТБ СибГИУ. – Электрон. дан. – Новокузнецк, [199-]. – Режим доступа: <http://libr.sibsiu.ru>, свободный. – Загл. с экрана.

2 Электронная библиотека СибГИУ [Электронный ресурс] : база данных содержит полнотекстовые электронные документы, поступающие в фонд НТБ СибГИУ. – Электрон. дан. – Новокузнецк, [200-]. – Режим доступа: <http://library.sibsiu.ru/LibrELibraryFullText.asp>. – Загл. с экрана.

3 Университетская библиотека online [Электронный ресурс] : электронно-библиотечная система / ООО «Директ-Медиа». – Москва, [200-]. – Режим доступа: <http://www.biblioclub.ru>. – Загл. с экрана.

4 Лань [Электронный ресурс] : электронно-библиотечная система / ООО «Издательство Лань». – Электрон. дан. – Санкт-Петербург, [200-]. – Режим доступа: <http://e.lanbook.com>. – Загл. с экрана.

5 Консультант студента. Электронная библиотека технического ВУЗа [Электронный ресурс] : электронно-библиотечная система / ООО «Политехресурс». – Электрон. дан. – Москва, [200-]. – Режим доступа: <http://www.studentlibrary.ru>. – Загл. с экрана.

6 Юрайт. Электронная библиотека [Электронный ресурс] : электронно-библиотечная система / ООО «Электронное издательство Юрайт». – Москва, [200-]. – Режим доступа: <http://www.biblio-online.ru>. – Загл. с экрана.

7 Научная электронная библиотека eLIBRARY.RU : электронное периодическое издание / ООО «РУНЭБ». – Электрон. дан. – Москва, [200-]. – Режим доступа: <http://elibrary.ru>. – Загл. с экрана.

8 Университетская информационная система РОССИЯ (УИС РОССИЯ) [Электронный ресурс] : электронная библиотека / НИВЦ МГУ им. М.В. Ломоносова. – Электрон. дан. – Москва, [200-]. – Режим доступа: <http://uisrussia.msu.ru>. – Загл. с экрана.

#### **г) программное обеспечение:**

- Microsoft Visual Studio 2010,
- NetBeans IDE 8.0 (свободно распространяемое ПО),
- Kaspersky Endpoint Security,
- «Программное обеспечение «Руконтекст»,
- WinRAR 3.6 и 7-Zip (свободно распространяемое ПО),
- Microsoft Office 2007,
- Microsoft Windows 7.

#### **д) информационно-справочные системы:**

1 Техэксперт [Электронный ресурс] : информационно-справочная система / ООО «Кузбасский центр нормативно-технической документации». – Электрон. дан. – Кемерово, [200-]. – Режим доступа: компьютерная сеть Сиб. гос. индустр. ун-та.

2 Консультант Плюс [Электронный ресурс] : справочно-правовая система / ООО «Информационный центр АНВИК». – Электрон. дан. – Москва, [199-]. – Режим доступа: компьютерная сеть библиотеки Сиб. гос. индустр. ун-та.

3 ГАРАНТ [Электронный ресурс] : справочно-правовая система / ООО «НПП «ГАРАНТ-СЕРВИС». – Электрон. дан. – Москва, [200-]. – Режим доступа: компьютерная сеть библиотеки Сиб. гос. индустр. ун-та.

4 Электронный реферативный журнал (ЭлРЖ) [Электронный ресурс] : база данных / ВИНТИ РАН. – Электрон. дан. – Москва, [200-]. – Режим доступа: компьютерная сеть библиотеки Сиб. гос. индустр. ун-та.

## **9 Материально-техническое обеспечение учебной дисциплины**

Материально-техническое обеспечение учебной дисциплины «Программная инженерия» включает специально оборудованный компьютерный класс с выходом в Интернет, аудиторию с оборудованным мультимедийным проектором, научно-техническую библиотеку СибГИУ.

## **10 Методические рекомендации по организации изучения учебной дисциплины**

Текущий контроль успеваемости обучающихся по учебной дисциплине «Программная инженерия» проводится в форме аттестации на основе оценки выполнения практических работ, курсовой работы, контроля за посещаемостью и т.п.

Промежуточная аттестация обучающихся по дисциплине «Программная инженерия» проводится в форме дифференцированного зачета по курсовой работе и экзамена на основе оценки результатов ответов обучающихся на теоретические вопросы, составленные по всем разделам изучаемой учебной дисциплины и выполнения практического задания.

Программа составлена в соответствии с требованиями ФГОС ВО и с учетом ООП по направлению подготовки 09.03.03 Прикладная информатика (направленность «Прикладная информатика в информационной сфере»).

Составитель:

к.т.н., доц. каф. ПИТиП

Н.Б. Бабичева

Программа рассмотрена и утверждена на заседании кафедры прикладных информационных технологий и программирования, протокол № 13 от «13» марта 2018 г.

зав. кафедрой ПИТиП

С.П. Огнев

Согласовано:

старший методист  
методического отдела

## Приложение А

**Аннотация**  
**программы учебной дисциплины «Программная инженерия»**  
**по направлению подготовки**  
**09.03.03 – Прикладная информатика**  
**направленность «Прикладная информатика в информационной**  
**сфере»**  
**форма обучения – заочная**

### 1 Цели и задачи освоения учебной дисциплины

В настоящее время происходит интенсивное развитие подходов и парадигм программирования, возникают и развиваются языки программирования. Поэтому профессионалы в области информационных технологий должны обладать навыками свободного обращения с компьютером, владения рядом прикладных программ и операционных систем, программирования на языках программирования высокого уровня. А также профессионалы в области информационных технологий должны знать принципы и особенности современных подходов к программированию, одним из самых актуальных подходов является объектно-ориентированный. Знание технологии проектирования, разработки, тестирования и эксплуатации программного продукта и умение применить свои знания на практике способствует становлению зрелого мышления профессионала по информационным технологиям и хорошему знанию методов современного программирования, что положительным образом скажется на его дальнейшем профессиональном самоопределении.

Основной **целью** курса является формирование у обучающихся общей культуры использования современных подходов к программированию, основываясь на современных требованиях информационных технологий. Понимание основных принципов, технологий построения программного обеспечения информационных систем общего назначения позволит выпускнику лучше ориентироваться и должным образом проявить себя на современном рынке информационных технологий.

**Задачами** курса являются:

- систематическое изложение лекционного материала по принципам применения современных процессов проектирования, разработки, тестирования и эксплуатации программного продукта и о взаимосвязи всех аспектов программной инженерии;
- изучение и сравнительный анализ современных процессов проектирования и разработки программных продуктов;
- изучение принципов и методов оценки качества и управления качеством программного продукта;

- формирование практических навыков по созданию программ на основе объектно-ориентированного подхода с использованием различных языков программирования;
- приобретение практических навыков формирования и анализа требований, оценки качества и тестирования программных продуктов.

## **2 Место учебной дисциплины в структуре ООП по направлению подготовки**

Дисциплина «Программная инженерия» входит в вариативную часть (раздел «Обязательные дисциплины») учебного плана бакалавров и является обязательной для изучения.

Для успешного освоения дисциплины обучающиеся должны иметь практические навыки создания программ с использованием языков программирования высокого уровня.

Для успешного изучения дисциплины обучающиеся используют знания и умения, сформированные в ходе изучения дисциплин «Содержательные основы прикладной информатики и вычислительной техники», «Основы программирования», «Программирование».

Изучение данной дисциплины является основой для последующего изучения дисциплин «Инфокоммуникационные системы и сети», «Информационная безопасность и защита информации», «Web-технологии», «Web-программирование», «Проектирование информационных систем», а также для прохождения практик, выполнения курсовых проектов и работ, последующей подготовки к государственной итоговой аттестации.

## **3 Планируемые результаты обучения по учебной дисциплине**

Процесс изучения учебной дисциплины «Программная инженерия» направлен на формирование следующих компетенций:

### **– профессиональные компетенции (ПК):**

*ПК-1* - способностью проводить обследование организаций, выявлять информационные потребности пользователей, формировать требования к информационной системе.

Структура компетенции:

- знать: экономико-правовые основы разработки программных продуктов;

- уметь: формулировать требования к создаваемым программным комплексам;

- владеть: навыками разработки программных комплексов для решения прикладных задач, оценки сложности алгоритмов и программ, использования современных технологий программирования, тестирования и документирования программных комплексов.

*ПК-2* - способностью разрабатывать, внедрять и адаптировать прикладное программное обеспечение.

Структура компетенции:

- знать: задачи и методы исследования и обеспечения качества и надежности программных компонентов и систем;
- уметь: использовать международные и отечественные стандарты;
- владеть: навыками разработки программных комплексов для решения прикладных задач, оценки сложности алгоритмов и программ, использования современных технологий программирования, тестирования и документирования программных комплексов.

*ПК-8* - способностью программировать приложения и создавать программные прототипы решения прикладных задач.

Структура компетенции:

- знать: принципы организации проектирования, разработки и содержания этапов процесса разработки программных комплексов;
- уметь: применять современные подходы и технологии программирования;
- владеть: навыками разработки программных комплексов для решения прикладных задач, оценки сложности алгоритмов и программ, использования современных технологий программирования, тестирования и документирования программных комплексов.

**– профессионально-специализированные компетенции (ПСК):**

*ПСК-1* - способностью проектировать программные комплексы, базы данных, автоматизированные информационные системы на основе современных инструментальных средств и технологий программирования.

Структура компетенции:

- знать: профили открытых информационных систем, функциональные и технологические стандарты разработки программных комплексов;
- уметь: формировать архитектуру программных комплексов для информатизации предприятий, разрабатывать программные приложения;
- владеть: навыками разработки программных комплексов для решения прикладных задач, оценки сложности алгоритмов и программ, использования современных технологий программирования, тестирования и документирования программных комплексов.

#### **4 Структура и содержание учебной дисциплины**

Дисциплина проводится в течение третьего курса.

Общая трудоемкость дисциплины «Программная инженерия» (3 курс) составляет 5 зачетных единиц (180 академических часов), из них 24 аудиторных часа и 156 часов на самостоятельную работу (включая 9 часов на подготовку к экзамену).

#### **5 Краткое содержание учебной дисциплины**

В структуре дисциплины выделяются следующие основные разделы (темы):

Раздел 1. Технология разработки программного средства: 1.1. Определение программной инженерии. 1.2. Методы проектирования архитектуры системы. 1.3. Методы спецификации. 1.4. Интерфейсы.

Раздел 2. Управление требованиями и качеством: 2.1. Методы определения требований. 2.2. Методы анализа требований. 2.3. Характеристики качества программного обеспечения. 2.4. Метрики качества. 2.5. Сложность программных средств. 2.6. Корректность программных средств. 2.7. Надежность программных средств.

Раздел 3. Тестирование программного продукта: 3.1. Виды и методы тестирования. 3.2. Эвристические методы создания тестов. 3.3. Документирование тестирования. 3.4. Автоматизация тестирования.

## **6 Формы организации учебного процесса**

Программой учебной дисциплины «Программная инженерия» предусмотрено проведение лекций и практических занятий, руководство курсовой работой. Особое место в овладении учебной дисциплины «Программная инженерия» отводится самостоятельной работе, позволяющей получить максимальное представление о данной учебной дисциплине.

Контактная работа обучающихся с преподавателем включает в себя занятия лекционного типа (лекции) практические занятия, групповые консультации, индивидуальную работу обучающихся с преподавателем, промежуточную аттестацию обучающихся и иную контактную работу, предусматривающую групповую или индивидуальную работу обучающихся с преподавателем. Контактная работа обучающихся с преподавателем может быть аудиторной, внеаудиторной, а также проводиться в электронной информационно-образовательной среде.

## **7 Виды промежуточной аттестации**

Экзамен и дифференциальный зачет по курсовой работе.

## **8 Составитель:**

к.т.н., доц. каф. ПИТиП Н.Б. Бабичева



**Дополнения и изменения к программе учебной дисциплины «Программная инженерия»  
основной образовательной программы  
09.03.03 – Прикладная информатика  
направленность «Прикладная информатика в информационной сфере»  
на период 2018 – 2023 г.г.**

Номер изменения/дополнения	Содержание дополнения/изменения	Основание внесения изменения/дополнения
		протокол заседания кафедры № _____ от «___» _____ 20__ г.
		протокол заседания кафедры № _____ от «___» _____ 20__ г.
		протокол заседания кафедры № _____ от «___» _____ 20__ г.
		протокол заседания кафедры № _____ от «___» _____ 20__ г.
		протокол заседания кафедры № _____ от «___» _____ 20__ г.