

МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РОССИЙСКОЙ ФЕДЕРАЦИИ

ПЕНЗЕНСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ

ФАКУЛЬТЕТ ВЫЧИСЛИТЕЛЬНОЙ ТЕХНИКИ

УТВЕРЖДАЮ  
Декан ФВТ  
Л.Р. Фионова  
« 10 » сентября 2015 г.



## РАБОЧАЯ ПРОГРАММА ДИСЦИПЛИНЫ

### Б1.1.23 Программная инженерия

Направление подготовки – *09.03.03 Прикладная информатика*

Профиль подготовки – *Прикладная информатика в экономике*

Квалификация (степень) выпускника – *бакалавр*

Форма обучения – *заочная*

г. Пенза, 2015 г.

## 1. Цели освоения дисциплины

Целями освоения дисциплины «Программная инженерия» приобретение студентами теоретических знаний и практических навыков создания программных продуктов, в частности, автоматизированных систем, основанных на современных технологиях их проектирования и разработки.

**Задачи** изучения дисциплины:

- изложение системы основных концепций и понятий, используемых в современных технологиях разработки программных систем;
- описание основных технологий разработки программных систем;
- ознакомление с основными инструментальными средами разработки программных систем;
- приобретение навыков работы в командах разработчиков;
- развитие логического мышления, навыков исследования явлений и процессов, связанных с предметной деятельностью;
- формирование навыков самостоятельной работы, организации исследовательской работы.

## 2. Место дисциплины в структуре ОПОП бакалавриата

Дисциплина «Программная инженерия» входит в базовую часть блока Б1.1 Дисциплины Б1. Ее изучение базируется на знаниях, приобретенных студентами в ходе изучения предшествующих дисциплин: «Математика», «Теория вероятностей и математическая статистика», «Информационные системы и технологии», «Базы данных», «Проектный практикум», «Информатика и программирование».

Получаемые при изучении дисциплины знания и умения найдут применение при выполнении студентами производственной практики и выпускной бакалаврской работы.

## 3. Компетенции обучающегося, формируемые в результате освоения дисциплины «Программная инженерия»

Процесс изучения дисциплины направлен на формирование элементов следующих компетенций в соответствии с ФГОС ВО по данному направлению:

Коды компетенции	Наименование компетенции	Структурные элементы компетенции (в результате освоения дисциплины обучающийся должен знать, уметь, владеть)
1	2	3
ПК-2	способность разрабатывать, внедрять и адаптировать прикладное программное обеспечение	Знать: назначение и виды ИС; состав функциональных и обеспечивающих подсистем ИС; модели и процессы жизненного цикла ИС; виды угроз ИС и методы обеспечения информационной безопасности
		Уметь: выполнять работы на всех стадиях жизненного цикла проекта ИС, выявлять угрозы информационной безопасности, обосновывать организационно-технические мероприятия по защите информации в ИС
		Владеть: основами эксплуатации и сопровождения информационных систем и сервисов
ПК-3	способность проектировать ИС в соответствии с профилем подготовки по	Знать: принципы организации проектирования и содержание этапов процесса разработки ИС; канонические модели языка UML

	видам обеспечения	<p>Уметь: проектировать ИС в стандарте языка UML</p> <p>Владеть: навыками проектирования ИС для решения прикладных задач</p>
ПК-4	способностью документировать процессы создания информационных систем на стадиях жизненного цикла	Знать: состав и структуру проектной документации
		Уметь: документировать процессы создания ИС
		Владеть: навыками разработки пользовательской документации
ПК-8	способностью программировать приложения и создавать программные прототипы решения прикладных задач	Знать: основные идеи, принципы и методы структурного программирования, правила записи структурированных алгоритмов и программ, базовые управляющие структуры.
		Уметь: использовать операторы цикла для программирования повторяющихся действий; определять массивы, структуры и объединения, разрабатывать алгоритмы и программы с использованием структурированных типов данных.
		Владеть: технологией разработки отладки на алгоритмическом языке высокого уровня в среде современной инструментальной системы визуального программирования.
ПК-12	способностью проводить тестирование компонентов программного обеспечения ИС	Знать: методы и виды тестирования, критерии качества тестирования программного обеспечения, методику тестирования программных модулей, возможности тестирования модулей без исполнения программ, особенности тестирования комплектов программ, инструментальные средства отладки, ручные методы тестирования, автоматические методы тестирования
		Уметь: выполнять отладку и тестирование компонентов программного обеспечения ИС
		Владеть: применять методы и виды тестирования программ, использовать инструментальные средства тестирования
ПК-15	способностью осуществлять тестирование компонентов информационных систем по заданным сценариям	Знать: виды и методы тестирования программных средств
		Уметь: выполнять тестирование программных средств по различным сценариям
		Владеть: навыками формирования отчетов по результатам тестирования программных средств

#### 4. Структура и содержание дисциплины Программная инженерия

##### 4.1. Структура дисциплины

Общая трудоемкость дисциплины составляет **5** зачетных единиц, **180** часов.

№ п/п	Наименование разделов и тем дисциплины (модуля)	Семестр	Виды учебной работы, включая самостоятельную работу студентов и трудоемкость (в часах)									Формы текущего контроля успеваемости (по неделям семестра)							
			Аудиторная работа				Самостоятельная работа					Собеседование	Коллоквиум	Проверка тестов	Проверка лаборат. работ	Проверка реферата	Проверка эссе и иных творческих работ	курсовая работа (проект)	др.
			Всего	Лекция	Практические занятия	Лабораторные занятия	Всего	Подготовка к аудиторным занятиям	Реферат, эссе и др.	Контрольная работа (проект)	Подготовка к экзамену								
1.	<b>Тема 1.</b> Сущность программной инженерии	6	<b>6</b>	2		4	<b>25</b>	25											
2.	<b>Тема 2.</b> Модели жизненного цикла	6	<b>6</b>	2		4	<b>25</b>	25											
3.	<b>Тема 3.</b> Язык структурного моделирования	6	<b>6</b>	2		4	<b>25</b>	25											
4.	<b>Тема 4.</b> Язык объектно-ориентированного моделирования	6	<b>6</b>	2		4	<b>25</b>	25											
	<i>Контрольная работа</i>						<b>20</b>			20									
	<i>Подготовка к экзамену</i>						<b>36</b>				36								
	Общая трудоемкость, в часах		<b>24</b>	<b>8</b>		<b>16</b>	<b>156</b>	<b>100</b>		<b>20</b>	<b>36</b>	Промежуточная аттестация							
												Форма		Семестр					
												Зачет							
												Экзамен		6					

## 4.2. Содержание дисциплины Программная инженерия

### 4.2.1 Содержание лекционных разделов дисциплины

#### **Тема 1. Сущность программной инженерии**

Предпосылки и история дисциплины (Краткий обзор причин, истории возникновения и становления программной инженерии. Краткая характеристика основных методов проектирования программ, сложившихся в процессе поиска решения основной проблемы программной инженерии - сокращения стоимости программного обеспечения)

Программная инженерия - основные определения.

Стандартизация и стандарты Программной инженерии.

#### **Тема 2. Модели жизненного цикла**

Жизненный цикл программного обеспечения. Распределение финансовых и временных затрат на реализацию каждого из этапов разработки программного обеспечения. Обследование системы, общение с заказчиком, планирование разработки, составление технического задания.

Детальный анализ предметной области, принятие окончательного решения о необходимости создания информационной системы, проектирование общей архитектуры системы, выбор метода проектирования.

#### **Тема 3. Язык структурного моделирования**

Моделирование потока данных. Моделирование сущностей и отношений.

#### **Тема 4. Язык объектно-ориентированного моделирования**

Задачи и особенности объектно-ориентированного проектирования программных средств. Основные понятия и модели объектно-ориентированного проектирования программных средств. Варианты представления моделей и средства объектно-ориентированного проектирования программных средств. Идентификация классов и объектов. Основные результаты объектно – ориентированного проектирования (в стандарте UML): диаграммы классов, объектов, компонентов, состояний, последовательностей, действий. CASE – инструменты.

### 4.2.2 Перечень и содержание лабораторных занятий.

1 Разработка модели программной системы на основе использования объектного подхода и CASE – средств разработки.

Создание основных моделей (диаграмма вариантов использования, диаграмма классов, диаграмма последовательности, диаграмма деятельности) разрабатываемой программной системы в среде StartUML в соответствии с заданием.

Пример задания:

#### **Текстовый редактор**

Приложение предназначено для создания, обработки, форматирования текста (текстовых документов), должно поддерживать следующие функции:

- открытие нескольких документов;
- сохранение/загрузка файла;
- возможность изменять шрифт (тип, размер, начертание, цвет);
- поиск и замена текста;
- возможность выравнивания текста;
- возможность перемещения и/или копирования частей текста с помощью мыши (drag'n drop).

Интерфейс приложения должен иметь схожие черты с интерфейсами широко распространенных редакторов текстов (например, Word) и позволять осуществлять открытие файлов, их сохранение, работу с фрагментами текста (копирование, вставка, удаление) и т.п.

2 Реализация и испытание программной системы.

Среда реализации – Delphi.

Приложение должно работать в среде Windows XP/ Vista/ 7/ 8.

Комплексная отладка системы.

#### **4.2.3 Тематика и содержание контрольной работы**

В ходе выполнения контрольной работы необходимо разработать конфигурацию в среде Delphi и выполнить следующие этапы:

1. Разработка и утверждение технического задания
2. Разработка требований к программной системе
3. Разработка модели программной системы
4. Программирование и отладка
5. Оформление пояснительной записки

Пример задания на контрольную работу

Тема работы *Разработка системы «Расписание занятий студентов»*

*Приложение должно осуществлять следующие основные функции:*

*– добавление нового расписания;*

*– редактирование существующего расписания;*

*– просмотр расписания студентов заданного факультета и группы;*

*Разработать интерфейс приложения.*

Перечень заданий для контрольной работы приведен в приложении 1.

### **5. Образовательные технологии**

В процессе изучения дисциплины применяются следующие образовательные технологии:

- Лекции с применением мультимедиа технологий. Представление с помощью компьютерного проектора материала по темам № 2, 3.

- Лекции с проблемной постановкой: «Сущность программной инженерии», «Программная инженерия больше, чем программирование» тема № 1.

- Интерактивный разбор ситуаций – тема № 4.

Эффективности самостоятельной работы студентов будет способствовать использование технологий организации самостоятельной работы обучающихся, в основе которой – технология поиска и сбора новой информации; технология анализа новой информации; технология представления информации.

При реализации образовательных технологий используются следующие виды самостоятельной работы:

- 1) выполнение практических упражнений;
- 2) выполнение творческих заданий аналитического характера;
- 3) поиск информации в сети «Интернет», учебной и справочной литературе;
- 4) подготовка к сдаче экзамена.

**6. Учебно-методическое обеспечение самостоятельной работы студентов.  
Оценочные средства для текущего контроля успеваемости,  
промежуточной аттестации по итогам освоения дисциплины.**

При организации самостоятельной работы занятий используются следующие образовательные технологии: работа с научной, учебной и учебно-методической литературой, работа в сети Интернет для поиска необходимой информации.

Программой дисциплины предусмотрены следующие виды текущего контроля: выполнение студентами самостоятельных индивидуальных и групповых заданий.

По данной дисциплине предусмотрена форма отчетности – экзамен в 1 семестре.

При организации самостоятельной работы занятий используются следующие образовательные технологии:

1) проектная образовательная технология (подбор материалов, оформление их в виде тематического проекта, защита проектов) – **3, 4.**

2) проблемно - поисковый метод (работа с научной, учебной и учебно-методической литературой, работа в сети Интернет для поиска необходимой информации) – темы **1-2.**

На лабораторных занятиях разбираются типовые задания. Аналогичные задания более высокой сложности предлагаются для самостоятельного решения. По самостоятельным работам оформляется отчет.

### 6.1. План самостоятельной работы студентов

№ нед.	Тема	Вид самостоятельной работы	Задание	Рекомендуемая литература	Количество часов
1	<b>Тема 1. Сущность программной инженерии</b>	Проработка лекций; чтение обязательной и дополнительной литературы	Задание: используя учебную литературу и материалы сети Интернет, ответьте на следующие вопросы: <ol style="list-style-type: none"> <li>1. Что такое промышленный программный продукт. Дать определения пакета прикладных программ, программной системы.</li> <li>2. Жизненный цикл программного обеспечения. Дать краткую характеристику каждого этапа.</li> <li>3. Почему программные системы сложны. Привести пять признаков сложной системы.</li> <li>4. Техническое задание. Перечислить и охарактеризовать разделы, входящие в техническое задание.</li> </ol>	1, 2, 5*	25
2	<b>Тема 2. Модели жизненного цикла</b>	Поиск дополнительных электронных ресурсов по тематике лекций	Задание: используя учебную литературу и материалы сети Интернет подготовить ответы на следующие вопросы: <ol style="list-style-type: none"> <li>1. Что такое жизненный цикл программного обеспечения?</li> <li>2. Перечислите этапы жизненного цикла программного обеспечения.</li> <li>3. Перечислите основные подходы программирования</li> </ol>	2, 3, 4, 1*	25
3	<b>Тема 3. Язык структурного моделирования</b>	Проработка лекций; чтение обязательной и дополнительной литературы	Задание: используя учебную литературу и материалы сети Интернет подготовить рефераты на темы: <ol style="list-style-type: none"> <li>1. Опишите структурный подход программирования.</li> </ol>	1*	25



			<ol style="list-style-type: none"> <li>2. Опишите процедурный подход программирования.</li> <li>3. Опишите объектно-ориентированный подход программирования.</li> <li>4. Дайте определение процедуры и функции.</li> <li>5. Дайте определения программирования.</li> <li>6. Дайте определение структуры данных.</li> </ol>		
4	<b>Тема 4. Язык объектно-ориентированного моделирования</b>	Поиск дополнительных электронных ресурсов по тематике	<p>Задание: используя учебную литературу и материалы сети Интернет, ответьте на следующие вопросы:</p> <ol style="list-style-type: none"> <li>1. Что такое поток управления и поток данных?</li> <li>2. Что такое объект? Что может быть объектом?</li> <li>3. Что такое класс? Чем отличается класс от объекта?</li> <li>4. Чем характеризуется класс?</li> <li>5. Что такое состояние класса?</li> <li>6. Что такое поведение класса?</li> <li>7. Перечислите основные типы операций.</li> <li>8. Что такое «отношения между классами»?</li> <li>9. Что такое модель и моделирование?</li> <li>10. Что такое «классификация»?</li> </ol>	4, 2*, 3*, 4*	25

## 6.2. Методические указания по организации самостоятельной работы студентов

Текущая и опережающая самостоятельная работа (СРС) студентов направлена на углубление и закрепление знаний, а также практических умений. К ней относятся:

- изучение тем, вынесенных на самостоятельную проработку;
- работа бакалавров с лекционным материалом;
- поиск и анализ литературы и электронных источников информации по вопросам и проблемам, поставленным на лекционных занятиях;
- изучение методических указаний к лабораторным работам;
- подготовка к защите выполненных лабораторных работ;
- поиск и анализ литературы и электронных источников информации по подготовке рефератов;
- подготовка к экзамену.

## 6.3. Материалы для проведения текущего и промежуточного контроля знаний студентов

### *Контроль освоения компетенций*

№ п/п	Вид контроля	Контролируемые темы (разделы)	Компетенции, компоненты которых контролируются
1	Развернутый план-конспект (темы, вопросы), подготовка к экзамену	<b>Тема 1.</b> Сущность программной инженерии	ПК-2, ПК-3
		<b>Тема 2.</b> Модели жизненного цикла	ПК-2, ПК-4
		<b>Тема 3.</b> Язык структурного моделирования	ПК-12, ПК-15
		<b>Тема 4.</b> Язык объектно-ориентированного моделирования	ПК-8, ПК-12
2	Проверка лабораторных работ	Лабораторная работа №1-Лабораторная работа №9	ПК-8, ПК-12, ПК-15

Защита лабораторных работ предполагает представление оформленных по ним отчетов и демонстрацию работы.

### *Примерный перечень вопросов и заданий к экзамену*

1. Диаграмма вариантов использования (use case diagram). Основные элементы. Отношения.
2. Диаграмма пакетов. Импорт пакетов. Слияние пакетов.
3. Диаграмма классов (class diagram). Основные элементы. Отношения.
4. Диаграмма композитной структуры (composite structure diagram). Основные элементы.
5. Диаграмма объектов. Экземпляр.
6. Диаграмма деятельности (activity diagram). Узлы управления. Узлы потока объектов.
7. Диаграмма последовательности (sequence diagram). Линия жизни. Комбинированные фрагменты. Сообщения.
8. Диаграмма коммуникации (communication diagram). Линия жизни. Сообщения.
9. Диаграмма компонентов (component diagram). Компонент. Интерфейс. Порт. Отношения.

10. Диаграмма конечного автомата (state machine diagram). Простое состояние. Псевдосостояния. Переходы. Композитные состояния и регионы.
11. Диаграмм развертывания (deployment diagram). Узел. Артефакт. Отношения.
12. Этапы разработки программного обеспечения. Персонал. Процесс. Проект. Продукт. Качество.
13. Методы разработки программного обеспечения. Требования.
14. Разновидности процесса разработки. Водопадная модель.
15. Разновидности процесса разработки. Спиральная модель.
16. Разновидности процесса разработки. Инкрементальная модель.
17. Унифицированный процесс разработки программного обеспечения (USDP).
18. Процесс контроля качества. Метрики. Методы «белого» и «черного ящика». Инспектирование.
19. Индивидуальный процесс разработки (PSP). Командный процесс разработки (TSP). Модель зрелости возможностей (CMM).
20. Основные параметры управления проектами. Схема процесса.
21. Управление персоналом. Варианты организации персонала. Подбор участников проекта.
22. Риски. Управление рисками. Выявление и предупреждение.
23. Оценка стоимости проекта. Конструктивная модель стоимости проекта (COCOMO).
24. Оценка стоимости проекта. Метод определения функционального размера.
25. Анализ требований. С-требования.
26. Анализ требований. D-требования. Функциональные требования. Нефункциональные требования. Прослеживание.

## **7. Учебно-методическое и информационное обеспечение дисциплины Программная инженерия**

### а) основная литература:

1. Малышева Е.Н. Проектирование информационных систем Раздел 5. Индустриальное проектирование информационных систем. Объектно-ориентированная case-технология проектирования информационных систем. - Кемеровский государственный институт культуры, 2009. – 70. - [https://e.lanbook.com/book/49647#book\\_name](https://e.lanbook.com/book/49647#book_name)
2. Александров Д.В. Инструментальные средства информационного менеджмента. CASE-технологии и распределенные информационные системы. - Издательство "Финансы и статистика", 2011. – 224. - [https://e.lanbook.com/book/5306#book\\_name](https://e.lanbook.com/book/5306#book_name)

### б) дополнительная литература:

1. Ехлаков Ю.П. Введение в программную инженерию.- Томский государственный университет систем управления и радиоэлектроники, 2012. – 14. - <https://e.lanbook.com/book/11418#authors>
2. Антамошкин О.А. Программная инженерия. Теория и практика: учебник. - Сибирский Федеральный Университет, 2012. – 247. - [https://e.lanbook.com/book/45709#book\\_name](https://e.lanbook.com/book/45709#book_name)
3. Абдулаев В.И. Программная инженерия: учебное пособие. - Поволжский государственный технологический университет, 2016. – 168. - [https://e.lanbook.com/book/92577#book\\_name](https://e.lanbook.com/book/92577#book_name)

### в) программное обеспечение и Интернет-ресурсы

1. Руководство к своду знаний по программной инженерии (the Guide to the Software Engineering Body of Knowledge) [электронный ресурс]:  
<http://www.computer.org/portal/web/swebok>
2. Основы Программной Инженерии (по SWEBOK) [электронный ресурс]:  
[http://swebok.sorlik.ru/software\\_engineering.html](http://swebok.sorlik.ru/software_engineering.html)

## **8. Материально-техническое обеспечение дисциплины (модуля)**

Чтение лекций по дисциплине проводится с использованием мультимедийного компьютерного проектора.

Лабораторные занятия проводятся в классе, оснащённом персональными компьютерами с операционной системой Windows XP/Windows Vista/Windows 7 и установленными программными продуктами StartUML, Delphi.

## Перечень типовых заданий к лабораторным работам по дисциплине “Программная инженерия ”

В каждом из предложенных вариантов требуется при помощи CASE-средства StarUML построить модель программного обеспечения.

Процесс создания модели должен проходить так, как это описано в лабораторных работах.

Должны быть выполнены следующие действия:

1. составление глоссария проекта;
2. создание модели вариантов использования;
3. анализ вариантов использования;
4. проектирование системы;
5. реализация системы.

После выполнения третьего этапа модель должна удовлетворять перечисленным ниже требованиям. Глоссарий проекта должен иметь вид таблицы и храниться в отдельном файле. На диаграммах вариантов использования каждое действующее лицо (actor) и вариант использования должны сопровождаться описанием. Эти описания должны быть составлены на русском языке. Описание действующего лица должно коротко (в одну-две строки) сообщать о роли данного лица. Описание варианта использования должно включать в себя пояснение, предусловие, потоки событий (основной и альтернативные, если таковые есть) и постусловие. Описания представляют собой либо присоединенные текстовые файлы, либо текст, введенный в поле Documentation спецификации соответствующего элемента диаграммы. Диаграммы взаимодействия, соответствующие потокам событий вариантов использования, должны содержать необходимые пояснения.

При проектировании системы требуется:

1. создать иерархию классов системы;
2. разместить классы по пакетам (использовать деление: пользовательский интерфейс – управление – данные; или другое в зависимости от постановки задачи);
3. связать объекты с классами, сообщения на диаграммах взаимодействия – с операциями;
4. каждый класс снабдить описанием, которое должно включать в себя краткое описание (ответственность класса), описание атрибутов в виде таблицы (имя, описание, тип), таблицу с описанием операций (имя, описание, сигнатура);
5. для классов указать стереотипы;
6. построить диаграммы классов системы, отображающие связи между классами;
7. для описания поведения экземпляров отдельных классов построить диаграммы состояний;
8. разработать (если это требуется вариантом задания) схему базы данных и отобразить ее на диаграмме «сущность – связь».

При реализации системы необходимо построить диаграммы компонентов для каждого пакета и для системы в целом. Также следует разработать диаграмму размещения. В зависимости от варианта задания диаграмма размещения должна показывать расположение компонентов в распределенном приложении или связи между встроенным процессором и устройствами. Должна быть произведена проверка корректности модели и автоматическая генерация кода средствами StarUML.

Ниже перечислены варианты заданий.

1. Цифровой диктофон
2. Торговый автомат
3. Табло на станции метро
4. Система автоматизации для пункта проката видеокассет

5. Мини-АТС
6. Телефон
7. Стиральная машина
8. Таксофон
9. Банкомат
10. Холодильник
11. Кодовый замок
12. Турникет метро
13. Система учета товаров
14. Библиотечная система
15. Интернет-магазин
16. Web-конференция
17. Каталог ресурсов Интернет
18. Будильник
19. Генеалогическое дерево
20. Телевизор
21. Система поддержки составления расписания занятий
22. Домофон

Рабочая программа дисциплины Программная инженерия составлена в соответствии с требованиями ФГОС ВО по направлению подготовки 09.03.03 - Прикладная информатика.

Программу составили:

1. Кузнецова Ольга Юрьевна, доцент

  
(Ф.И.О., должность, подпись)

Настоящая программа не может быть воспроизведена ни в какой форме без предварительного письменного разрешения кафедры-разработчика программы.

Программа одобрена на заседании кафедры «Информационно-вычислительные системы»

Протокол № 9

от «06» апрель 2015 года

Зав. кафедрой Ю.Н. Косников

  
(подпись)

Программа одобрена методической комиссией ФВТ факультета (института)

Протокол № 5

от «10» апрель 2015 года

Председатель методической комиссии ФВТ

  
(подпись)

Н.Н. Коннов

**Сведения о переутверждении программы на очередной учебный год и регистрации изменений**

Учебный год	Решение кафедры (№ протокола, дата, подпись зав. кафедрой)	Внесенные изменения	Номера листов (страниц)		
			замененных	новых	аннулированных
2016/ 2017	Пр. № 11 от 22.06.16г <i>Тихомиров</i>	Переутверждение без изменений			
2017/ 2018	Пр. № 14 от 27.06.17г <i>Тихомиров</i>	Введена дополнительная лекция у ЭВС	11		