

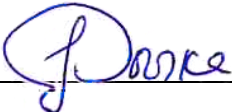
ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ ОБРАЗОВАТЕЛЬНОЕ
УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ
«КАМЧАТСКИЙ ГОСУДАРСТВЕННЫЙ ТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ»
(ФГБОУ ВО «КамчатГТУ»)

Факультет информационных технологий

Кафедра «Информационные системы»

УТВЕРЖДАЮ

Декан факультета информационных
технологий

 И.А. Рычка

«17» марта 2021 г.

РАБОЧАЯ ПРОГРАММА ДИСЦИПЛИНЫ

«Программная инженерия»

направление подготовки
09.03.03 Прикладная информатика
(уровень бакалавриата)

направленность (профиль):
«Прикладная информатика в экономике»

Петропавловск-Камчатский,
2021

Рабочая программа дисциплины составлена на основании ФГОС ВО по направлению подготовки 09.03.03 «Прикладная информатика», профиль «Прикладная информатика в экономике», учебного плана ФГБОУ ВО «КамчатГТУ».

Составитель рабочей программы

Доцент кафедры
«Информационные системы»


(подпись)

С.В. Чебанюк
(Ф.И.О.)

Доцент кафедры
«Информационные системы»

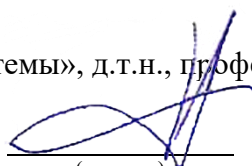

(подпись)

И.А. Рычка
(Ф.И.О.)

Рабочая программа рассмотрена на заседании кафедры «Информационные системы». «05» марта 2021 г., протокол №07.

Заведующий кафедрой «Информационные системы», д.т.н., профессор

«05» марта 2021 г.


(подпись)

И.Г. Проценко
(Ф.И.О.)

1. Цели и задачи учебной дисциплины

Целью освоения дисциплины «Программная инженерия» является изучение современных инженерных принципов (методов) создания надежного, качественного программного обеспечения, удовлетворяющего предъявляемым к нему требованиям; формирование у студентов понимания необходимости применения данных принципов программной инженерии.

Задачами изучения дисциплины «Программная инженерия» является получение обучающимися:

- формирование у студентов общего понимания структуры программной инженерии как отрасли научного и практического знания, актуальности программной инженерии как дисциплины, охватывающей все аспекты разработки программного обеспечения;

- формирование умений и навыков в области разработки ПО с прохождением всех стадий ЖЦ ПО с участием в разработке программной документации.

В результате изучения программы курса студенты должны:

Знать:

- процессы документирования программ и данных.

Уметь:

- документировать процессы создания информационных систем на стадиях жизненного цикла.

Владеть:

- приёмами документирования процессов создания информационных систем на стадиях жизненного цикла.

2. Требования к результатам освоения дисциплины

Процесс изучения дисциплины направлен на формирование следующих компетенций:

- способен участвовать в разработке стандартов, норм и правил, а также технической документации, связанной с профессиональной деятельностью (ОПК-4);

- способен устанавливать программное и аппаратное обеспечение для информационных и автоматизированных систем (ОПК-5).

Таблица – Планируемые результаты обучения при изучении дисциплины, соотнесенные с планируемыми результатами освоения образовательной программы

Коды формируемых компетенций	Планируемые результаты освоения образовательной программы	Планируемый результат обучения	Код показателя освоения
ОПК-4	Способен участвовать в разработке стандартов, норм и правил, а также технической документации, связанной с профессиональной деятельностью	Знать: – процессы документирования программ и данных.	З (ОПК-4) 1
		Уметь: – документировать процессы создания информационных систем на стадиях жизненного цикла.	У (ОПК-4)1
		Владеть: – приёмами документирования процессов создания информационных систем на стадиях жизненного цикла.	В (ОПК-4)1
ОПК-5	Способен устанавливать	Знать:	

	программное и аппаратное обеспечение для информационных и автоматизированных систем	процессы документирования программ и данных.	З (ОПК-5) 1
		Уметь: документировать процессы создания информационных систем на стадиях жизненного цикла.	У (ОПК-5)1
		Владеть: приёмами документирования процессов создания информационных систем на стадиях жизненного цикла.	В (ОПК-5)1

3. Место дисциплины в структуре образовательной программы

«Программная инженерия» является дисциплиной вариативной части в структуре образовательной программы.

Курс дисциплины «Программная инженерия» продолжает цикл дисциплин разработки информационных систем, опирается на знания в области информационных технологий и программирования, полученные на дисциплинах «Информатика и программирование», «Операционные системы», «Объектно-ориентированное программирование», «Базы данных», «Основы алгоритмизации и языки программирования» и во время прохождения учебной практики.

Знания, полученные обучающимися в процессе изучения дисциплины «Программная инженерия», позволят им выполнять на высоком уровне лабораторные и практические работы, курсовое и дипломное проектирование. Дисциплина «Программная инженерия» является базовой дисциплиной для курсов «Информационная безопасность», «Управление информационными системами» и выполнения дипломного проектирования.

4. Содержание дисциплины

4.1 Тематический план дисциплины

Наименование разделов и тем	Всего часов	Аудиторные занятия	Контактная работа по видам учебных занятий			Самостоятельная работа	Формы текущего контроля ¹	Итоговый контроль знаний по дисциплине
			Лекции	практические занятия	Лабораторные работы			
1	2	3	4	5	6	7	8	9
Раздел 1. Жизненный цикл программных средств	50	32	16	0	16	18	Опрос	
Основные понятия и определения программной инженерии	10	6	2		4	4	Опрос, РЗ	
Стандарты ЖЦ ПО	14	10	6		4	4	Опрос, РЗ	
Процессы ЖЦ ПО	12	8	4		4	4	Опрос, РЗ	
Документация в ЖЦ ПО	14	8	4		4	6	Опрос, РЗ	
Раздел 2. Системный анализ	58	40	20	0	20	18	Опрос	
Структурная методология анализа и построения спецификаций	14	8	4		4	6	Опрос, РЗ	

¹Т – тестирование, КС – кейс-стади, РЗ – решение задач, ПЗ – практические задания, Д - доклады

Технологии структурно-функционального анализа	22	16	8		8	6	Опрос, РЗ	
Объектно-ориентированная методология анализа и построения спецификаций	22	16	8		8	6	Опрос, РЗ	
Зачет							Опрос	
Раздел 3. Конструирование ПО	37	21	11	0	10	16	Опрос	
Системное проектирование программного продукта	6	2	2			4	Опрос	
Объектно-ориентированные модели архитектуры	12	8	4		4	4	Опрос, РЗ	
Правила организации пользовательского интерфейса приложений	9	5	3		2	4	Опрос, РЗ	
Технологии конструирования программного обеспечения	10	6	2		4	4	Опрос, РЗ	
Раздел 4. Тестирование и отладка программного продукта	26	13	6	0	7	13	Опрос	
Тестирование программного обеспечения	12	8	4		4	4	Опрос, РЗ	
Тестовая документация назначение	6	2	2			4	Опрос	
Отладка кода программного обеспечения	8	3			3	5	Опрос, РЗ	
КР	36					36		
Экзамен	45							45
Всего	252	106	53	0	53	101		45

3Ф0

Наименование разделов и тем	Всего часов	Аудиторные занятия	Контактная работа по видам учебных занятий			Самостоятельная работа	Формы текущего контроля ²	Итоговый контроль знаний по дисциплине
			Лекции	практические занятия	Лабораторные работы			
1	2	3	4	5	6	7	8	9
Раздел 1. Жизненный цикл программных средств	53	4	2	0	2		Опрос	
Основные понятия и определения программной инженерии	13	1	1			12	Опрос, РЗ	
Стандарты ЖЦ ПО	13	1	1			12	Опрос, РЗ	
Процессы ЖЦ ПО	13	1			1	12	Опрос, РЗ	
Документация в ЖЦ ПО	14	1			1	13	Опрос, РЗ	
Раздел 2. Системный анализ	49	4	2	0	2		Опрос	
Структурная методология анализа и построения спецификаций	16	1	1			15	Опрос, РЗ	
Технологии структурно-функционального анализа	17	2	1		1	15	Опрос, РЗ	
Объектно-ориентированная методология анализа и построения спецификаций	16	1			1	15	Опрос, РЗ	
Раздел 3. Конструирование ПО	56	4	2	0	2		Опрос	
Системное проектирование программного продукта	14	1	1			13	Опрос	
Объектно-ориентированные модели архитектуры	14	1			1	13	Опрос, РЗ	
Правила организации пользовательского интерфейса приложений	14	1			1	13	Опрос, РЗ	

²Т – тестирование, КС – кейс-стади, РЗ – решение задач, ПЗ – практические задания, Д - доклады

Технологии конструирования программного обеспечения	14	1	1			13	Опрос, РЗ	
Раздел 4. Тестирование и отладка программного продукта	49	4	2	0	2	13	Опрос	
Тестирование программного обеспечения	16	1	1			15	Опрос, РЗ	
Тестовая документация назначение	17	2	1		1	15	Опрос	
Отладка кода программного обеспечения	16	1			1	15	Опрос, РЗ	
КР	36					36		
Экзамен	9							9
Всего	252	16	8	0	8	227		9

4.2 Содержание дисциплины

Понятие программной инженерии. Процесс программной инженерии. Инструменты и методы программной инженерии. Качество ПО. Модели и профили жизненного цикла программных средств. Модели и процессы управления проектами программных средств. Архитектура ПО. Управление требованиями к ПО. Проектирование ПО. Конструирование ПО. Тестирование ПО. Сопровождение программного обеспечения. Конфигурационное управление. Зрелость управления процессами разработки ПО.

Основные серии стандартов, используемые в РФ в сфере ИТ и АС. Назначение стандартов серии ИСО 9000-14000. Основные организации, занимающиеся разработкой стандартов сфере ПО и ИТ. Основные категории программных документов. Состав технического задания на создание ПО.

Документирование ПО. Технико-экономическое обоснование проектов программных средств.

Раздел 1. Жизненный цикл программных средств

Тема 1.1 Основные понятия и определения программной инженерии

Эволюция общего программного обеспечения. Технология программирования. Основные этапы разработки ПО. Персонал, процесс, проект, продукт.

Эволюция системного программного продукта. Программный продукт и программа; системный программный продукт, программное средство, программное обеспечение; профессиональное программирование; информационные технологии и разработка программного продукта.

Тема 1.2 Стандарты ЖЦ ПО

Качество программного обеспечения и жизненный цикл. Назначение стандартов серии ИСО 9000-14000. Показатели качества ПО. Тотальное управление качеством TQM, принципы. Модель качества разработки ПО CMM, уровни зрелости компании в модели CMM. ISO/IEC 15504 (SPICE).

Тема 1.3 Процессы ЖЦ ПО

ЖЦ ПО стандарта ISO/IEC 12207, стадии, процессы. Модели ЖЦ ПО.

Руководство по Своду знаний по разработке программного обеспечения SWEBOOK V3.0.

Требования к ПО; проектирование ПО; конструирование ПО; тестирование ПО; сопровождение ПО; управление конфигурацией ПО.

Тема 1.4 Документация в ЖЦ ПО

Процессы документирования программ и данных. Технологическая документация процессов разработки и обеспечения всего жизненного цикла. Эксплуатационная документация программного продукта.

Проблемы организации документирования ПС. Формирование, структура и содержание документации, поддерживающей весь жизненный цикл ПС.

Документооборот в жизненном цикле проектов программных средств. ГОСТ Р ИСО/МЭК ТО 9294-93. Документирование в стандарте ISO 12207.

Стандартизация документирования процессов и продуктов ПС.

Лабораторная 1.1 Разработка приложения в RAD-среде.

Цель: актуализация знаний об основных этапах разработки ПО.

Рассматриваемые вопросы:

1. Технологии программирования.
2. Создание пакета приложения в RAD-среде.
3. Составление описания приложения.

Задания:

- Разработать структуру приложения и составить алгоритмы модулей и подпрограмм.
- Составить описание приложения.

Лабораторная 1.2 Качество программных средств

Цель: тестирование и оценка качественных показателей ПП.

Рассматриваемые вопросы:

1. Показатели качества программного обеспечения.
2. Методы определения показателей качества ПС.

Задания:

- Изучить ГОСТ 28.195-89 «Оценка качества программных средств. Общие положения».
- Исследовать заданный пример программы с целью определения ее эффективности и качества.
- Определить основные критерии оценки и количественные характеристики для заданной программы.
- Предложить варианты повышения эффективности и улучшения качества для заданного примера программы.

Лабораторная 1.3 Процессы разработки ПО

Цель: 1) обзор стандартов ЖЦ ПО; 2) описание модели ЖЦ ПО.

Рассматриваемые вопросы:

1. Достоинства и недостатки моделей ЖЦ ПО.
2. Выбор модели ЖЦ ИС.

Задания:

- Определить достоинства и недостатки моделей ЖЦ ИС
- Выбрать и обосновать выбор модели ЖЦ ИС для выполнения индивидуального проектного задания.

Лабораторная 1.4 Разработка технического задания на ИС

Цель: составление технической документации.

Рассматриваемые вопросы:

1. Структура технического задания на разработку АС.
2. Организация документирования.

Задания:

- Определить основные разделы технического задания на АС.
- Создать шаблон технического задания на АС.

Самостоятельная работа по разделу. Работа с конспектом лекций и рекомендованной литературой. Защита лабораторных работ.

Раздел 2. Системный анализ

Тема 2.1 Структурная методология анализа и построения спецификаций

Структурный подход к проектированию ИС. Структурная методология анализа и построения спецификаций. Структурно-функциональный подход. Методика построения функциональной модели. Модели «AS-IS» и «TO-BE». Диаграмма потоков данных..

Тема 2.2 Технологии структурно-функционального анализа

Технология структурно-функционального анализа (SADT).

Диаграммы функционального моделирования, синтаксис и применение диаграмм. Стандарты описания бизнес-процессов семейства IDEF. Состав IDEF0-модели. Порядок моделирования деятельности. Цикл разработки модели. Проверка диаграммы. Методика построения информационной модели. Диаграммы потоков данных, бизнес-процессы. Case-метод Баркера.

Тема 2.3 Объектно-ориентированная методология анализа и построения спецификаций

Унифицированный язык моделирования, UML и его место в анализе и проектировании программных систем. Основы процесса разработки. Начальная фаза проекта, уточнение, конструирование.

Диаграммы структуры и диаграммы поведения. Варианты использования и их диаграммы. Диаграммы классов, CRC-карты, диаграммы взаимодействия, пакетов, состояний, деятельностей, размещения. UML и программирование.

Лабораторная 2.1

Цели: 1) применение системы стандартов в области ПО, 2) применение методов функционального и системного моделирования, 3) построение структурных моделей бизнес.

Рассматриваемые вопросы:

1. Структурный анализ предметной области.
2. Диаграммы функционального моделирования.

Задания:

– выполнить структурный анализ предметной области и проектирование системы одним из структурно-функциональных подходов для задач по вариантам

Лабораторная 2.2

Цели: рассмотрение систем международных стандартов в области ПО и их применения при построении диаграмм описания документооборота и обработки информации.

Рассматриваемые вопросы:

1. Структурный анализ предметной области.
2. Диаграмма потоков данных и информационная модель.

Задания:

– построить диаграммы потоков данных для листовых функций.

Лабораторная 2.3

Цель: рассмотрение технологии описания бизнес-логики.

Рассматриваемые вопросы:

1. Описание бизнес-логики процессов предметной области.
2. Сценарий процесса и структура сопутствующего документооборота.

Задания:

– описать логику взаимодействия информационных потоков в нотации IDEF3.

Лабораторная 2.4

Цели: 1) применение методов объектно-ориентированного анализа; 2) построение диаграмм структуры и диаграмм поведения.

Рассматриваемые вопросы:

1. Нотация UML.
2. Диаграммы классов, диаграммы деятельности.

Задания:

– построить диаграммы классов и пакетов, продемонстрировать связи и ассоциации.
– построить диаграммы вариантов использования и диаграммы деятельности.

Самостоятельная работа по разделу. Работа с конспектом лекций и

рекомендованной литературой. Защита лабораторных работ.

Раздел 3. Конструирование ПО

Лекция 3.1 Системное проектирование программного продукта

Основы проектирования. Ключевые вопросы проектирования. Процесс объектно-ориентированного проектирования, преимущества ОО методологии.

Архитектура программного обеспечения, задачи, цели, классификация. Архитектурное представление (логическое представление, представление процесса, представление развертывания и представление реализации). Документ архитектуры программного обеспечения, разработчик архитектуры.

Архитектурные стили проектирования. Графическое представление архитектуры.

Лекция 3.2 Объектно-ориентированные модели архитектуры

Объектно-ориентированные модели системной архитектуры. Модели подсистем, Модели последовательности, Модели конечного автомата.

Модели реализации объектно-ориентированных программных систем. Компонентные диаграммы. Диаграммы размещения.

Лекция 3.3 Правила организации пользовательского интерфейса приложений

Пользовательский интерфейс и его роль в эффективности информационных систем. Психология восприятия человеком информации и принятия решений, ее учет при разработке программного обеспечения.

Стандарты пользовательских интерфейсов.

Разработка требований к пользовательскому интерфейсу и проектирование его реализации. Интеллектуальный интерфейс и его эргономика.

Инструментарий разработчика и его использование при создании информационных систем. Разработка требований к конструированию форм и разработка управляющих меню. Использование готовых компонентов в приложениях

Лекция 3.4 Технологии конструирования программного обеспечения

Аспекты конструирования программного обеспечения. Стандарты в конструировании. Языки конструирования. Кодирование, стиль хорошего программирования. Тестирование в конструировании. Повторное использование. Качество конструирования. Интеграция.

Управление конструированием ПО. Модели конструирования. Планирование конструирования. Измерения в конструировании.

Лабораторная 3.1

Цель: 1) логическое представление архитектуры ПО; 2) реализация проектных решений.

Рассматриваемые вопросы:

1. ГОСТ Р 57100-2016 Системная и программная инженерия. Описание архитектуры.
2. Нотация UML.
3. Диаграммы вариантов использования, диаграммы поведения.
4. Прототипирование.

Задания:

- построить диаграммы вариантов использования и диаграммы деятельности;
- составить документ архитектуры программного обеспечения.

Лабораторная 3.2

Цель: проектирование пользовательского интерфейса приложения.

Рассматриваемые вопросы:

1. Принципы организации пользовательского интерфейса.
2. Нотация UML.
3. Диаграммы вариантов использования.

Задания:

- Разработать прототип интерфейса windows-приложения в соответствии с основными принципами проектирования интерфейса.

Лабораторная 3.3

Цель: составление системной программной спецификации.

Рассматриваемые вопросы:

1. перечень требований и функциональных характеристик разрабатываемой программы.
2. разработка алгоритма решения задачи;
3. запись алгоритма.

Задания:

- разработать алгоритм решения конкретной задачи и представить его блок-схемой;
- составить документ «Техническое задание» на реализацию разработанного алгоритма.

Самостоятельная работа по разделу. Работа с конспектом лекций и рекомендованной литературой. Защита лабораторных работ.

Раздел 4. Тестирование и отладка программного продукта

Лекция 4.1 Тестирование программного обеспечения

Основные определения тестирования и отладки. Верификация и валидация. Цели тестирования. Этапы тестирования. Стратегия тестирования. Классификация ошибок. Градация серьезности дефекта. Градация приоритета дефекта. Уровни тестирования. Принципы тестирования.

Стандарты проведения тестирования в области информационных технологий. классификация тестирования. Методика тестирования: модульное тестирование, интеграционное тестирование, системное тестирование, приемочные испытания.

Тестирование методами черного и белого ящиков. Виды и типы тестирования.

Системное и регрессионное тестирование. Автоматизация тестирования. Инструменты отладки программного обеспечения. Методы отладки программного обеспечения.

Лекция 4.2 Тестовая документация назначение

Тестовая документация назначение, структура, особенности. Формирование и отслеживание реализации плана тестирования.

Техника граничных значений. Таблица принятия решений. Метод уникальных пар.

Рабочая и проектная документация тестировщика. Тест план. Чек-лист. Наборы тестовых сценариев. Баг-репорт. Отчет о тестировании.

Инструменты тестирования. Автокликеры. Postman. SoapUI. JMetр. Selenium.

Особенности тестирования веб-приложений

Системы документирования дефектов в области ПО

Лекция 4.3 Отладка кода программного обеспечения

Понятие отладки. Классификация методов отладки: протоколирование, работа в интерактивном отладчике, диагностика в процессе исполнения, использование графических средств, визуальный просмотр кода, анализ crash dump.

Классификация видов отладки: непосредственная и удалённая отладка.

Лабораторная 4.1

Цель: Запись текстов программ на алгоритмическом языке высокого уровня.

Рассматриваемые вопросы:

1. правила хорошего стиля программирования.
2. тест-план.
3. Запись текстов программ на алгоритмическом языке высокого уровня.

Задания:

- составить тест-план тестирования.
- написать программу на конкретном алгоритмическом языке программирования для решения конкретной задачи, используя приемы и методы программирования.

Лабораторная 4.2

Цель: тестирование и отладка программ.

Рассматриваемые вопросы:

1. правила хорошего стиля программирования.
2. протокол тестирования.
3. Структурное и функциональное тестирование.

Задания:

- составить набор тестов к разработанной ранее программе и провести ее отладку.
- оформить протокол тестирования.

Самостоятельная работа по разделу. Работа с конспектом лекций, рекомендованной литературой и ресурсами информационно-телекоммуникационной сети «Интернет». Защита лабораторных работ.

5 Учебно-методическое обеспечение для самостоятельной работы обучающихся

5.1. Внеаудиторная самостоятельная работа студентов

В целом внеаудиторная самостоятельная работа студента при изучении курса включает в себя следующие виды работ:

- проработка (изучение) материалов лекций;
- чтение и проработка рекомендованной основной и дополнительной литературы;
- подготовка к практическим занятиям;
- поиск и проработка материалов из Интернет-ресурсов, научных публикаций;
- выполнение домашних заданий в форме практических заданий, докладов и рефератов;
- выполнение курсового проектирования;
- подготовка презентаций для иллюстрации результатов курсового проектирования, докладов;
- подготовка к текущему(индивидуальные опросы) и итоговому (промежуточная аттестация) контролю знаний по дисциплине (экзамен).

Основная доля самостоятельной работы студентов приходится на подготовку к лабораторным занятиям, тематика которых полностью охватывает содержание курса. Самостоятельная работа по подготовке к лекционным занятиям предполагает умение работать с первичной информацией.

Самостоятельная работа по разделу 1:

Работа с конспектом лекций и рекомендованной литературой (1-3 и дополнительная).

Подготовка материалов к контрольному опросу по изученным темам, практических занятиях, диалогах с преподавателем и участниками проверки знаний дисциплинарного модуля.

Самостоятельная работа по разделу 2:

Работа с конспектом лекций и рекомендованной литературой (1-3 и дополнительная).

Подготовка материалов к контрольному опросу по изученным темам, практических занятиях, диалогах с преподавателем и участниками проверки знаний дисциплинарного

модуля.

Самостоятельная работа по разделу 3:

Работа с конспектом лекций и рекомендованной литературой (1-3 и дополнительная).

Подготовка материалов к контрольному опросу по изученным темам, практических занятиях, диалогах с преподавателем и участниками проверки знаний дисциплинарного модуля.

Самостоятельная работа по разделу 4:

Работа с конспектом лекций и рекомендованной литературой (1-3 и дополнительная).

Подготовка материалов к контрольному опросу по изученным темам, практических занятиях, диалогах с преподавателем и участниками проверки знаний дисциплинарного модуля.

6 Фонд оценочных средств для проведения промежуточной аттестации обучающихся по дисциплине

Фонд оценочных средств для проведения промежуточной аттестации обучающихся по дисциплине «Программная инженерия» представлен в приложении к рабочей программе дисциплины и включает в себя:

- перечень компетенций с указанием этапов их формирования в процессе освоения образовательной программы;
- описание показателей и критериев оценивания компетенций на различных этапах их формирования, описание шкал оценивания;
- типовые контрольные задания или материалы, необходимые для оценки знаний, умений, навыков и (или) опыта деятельности, характеризующих этапы формирования компетенций;
- методические материалы, определяющие процедуры оценивания знаний, умений, навыков и (или) опыта деятельности, характеризующих этапы формирования компетенций.

Вопросы для проведения промежуточной аттестации по дисциплине (экзамен)

1. Порядок разработки приложения. Через какие основные шаги разработки программы проходит приложение? Что такое «алгоритмизация»? Какая модель жизненного цикла программного обеспечения поддерживается RAD-технологией? Какие преимущества дает использование RAD-среды? Дать определение проекту. Особенности программного проекта. Примеры проектов.

2. Какие модели жизненного цикла программного обеспечения известны? Укажите достоинства и недостатки каждой модели.

3. Перечислить компоненты и описать базовую нотацию DFD–технологии. Какие данные и диаграммы являются входом технологической операции «Построение диаграммы потоков данных». Какие требования предъявляются к мини-спецификации? Этапы построения моделей в DFD–технологии: основные виды и последовательности работ.

4. Система обозначений в IDEF1x (компоненты SADT). Укажите назначение диаграмм потоков данных.

5. Какие основные «строительные блоки» используются для построения диаграмм IDEF3? В каких случаях строится перекресток ветвления «ИЛИ»? Сколько типов диаграмм в стандарте IDEF3?

6. Основные принципы ООА. Различия и общее в структурном и объектно-ориентированном подходах.

7. Дать определение терминов: «требования», «спецификация». Что подразумевается под «успех проекта». Характеристики превосходных требований. 5. Какой стандарт ЕСПД определяет содержание Технического задания? Назначение документа и его обязательные разделы. 6. Характеристика основных уровней стандартизации. Стандарты документирования ПО. Перечислите основные виды нормативных документов. Какие проблемы сопровождают внутрифирменные стандарты? 9. Схема классификации стандартов в области ИТ. Эволюция стандартов ПО. ЖЦ ПО. Эволюция ЖЦ ПО (по ISO/IEC 12207:1995). Процессы ЖЦ, регламентируемые стандартом ISO/IEC 12207.

8. Содержание государственного стандарта «Единая система программной документации».

9. Критерии качества ПО, факторы влияющие на качество ПО.

10. Дать определения для: Интерфейс, Глоссарий проекта, Интерфейс управления, Информационный интерфейс, Интерфейс ввода-вывода, Внешний интерфейс.

11. Принципы и правила проектирования интерфейса пользователя. Что значит «user-centered design» интерфейс? Система стандартов «Common user access» фирмы IBM

12. Уровни требований к ПО. Опишите вероятные риски появления «плохих» требований. Перечислите основные стандарты, регламентирующие состав документа «Спецификация требований»

13. Управление проектами. История управления проектами. Категории управления проектами. Треугольник ограничений проекта.

14. PMBOK: 9 областей управленческих знаний.

15. SQI: 34 компетенции IT менеджера

16. Управление командой проекта. Ролевая модель команды. Модели организации команд. Peopleware – человеческий фактор. Модели управления командой: административная модель, модель хаоса и модель открытой архитектуры. Общение в команде. Коммуникации. Принятие решений – компромисс и консенсус. Корпоративная политика.

17. Планирование и контроль. Задачи планирования. Как проверять и оценивать? Метрики проекта. Как надо планировать? Когда начинать планировать? СДР - структурная декомпозиция работ. Создание СДР. Критерии СДР. Стандарты планирования. 5. Средства управления проектом. Функции систем управления проектами. Обзор систем управления проектами.

18. Качество и управление качеством (экскурс в историю). Теория иерархии потребностей. 2. ISO9000: система управления качеством. Фундаментальные требования (TQM). Структура документов ISO9000. Как работает система управления качеством. Версии стандарта.

19. CMM: уровни зрелости процессов. Причины и история создания. Модель технологической зрелости. Пять уровней зрелости. Определение модели зрелости. Критерии оценки уровня зрелости.

20. ISO15504: аттестация, определение зрелости и усовершенствование процессов. Причины и история создания стандарта. Назначение и структура стандарта. Структура эталонной модели. Измерения «Процесс» и «Зрелость». Рейтинги атрибутов. Процесс аттестации

21. Правильность программ. Математическое доказательство правильности программ (верификация).

22. Тестирование. Цели тестирования. Виды тестирования. Принципы тестирования. Стратегия тестирования. Инспекции, сквозные просмотры и обзоры программ. Тестирование путем покрытия логики программы. Эквивалентное разбиение. Анализ граничных значений. Применение функциональных диаграмм. Предположение об

ошибке.

23. Основные цели процесса управления требованиями. Управление изменениями, контроль версий, контроль состояния требований, контроль за требованиями.

7 Рекомендуемая литература

7.1 Основная литература

1. Введение в программную инженерию: учеб. пособие / Е. В. Романова, С. В. Чебанюк, И. А. Рычка. - Петропавловск-Камчатский, КамчатГТУ, 2014. - 181 с.
2. Соммервилл, Иан. Инженерия программного обеспечения, 6-е издание.: Пер. с англ. – М.: Издательский дом Вильямс, 2002. – 624 с.
3. Иванова Г.С., Технология программирования: Учебник для вузов. – М.: Изд-во МГТУ им. Н.Э Баумана, 2002. – 320 с.

7.2 Дополнительная литература:

4. Стандартизация разработки программных средств : [Учеб. пособие для вузов по специальности 351400 "Прикладная информатика (в экономике)"] / В. А. Благодатских, В. А. Волнин, К. Ф. Посакалов ; Под ред. О. С. Разумова. - Москва : Финансы и статистика, 2003. - 284 с.
5. Структурные модели бизнеса: DFD-технологии / А.Н. Калашян, Г.Н. Калянов ; ред. Г.Н. Калянов . – М. : Финансы и статистика, 2003 . – 252 с.
6. Черемных С.В. и др. Моделирование и анализ систем. IDEF - технологии: практикум. – М. : Финансы и статистика. М., 2003. -208 с.
7. Липаев В.В. Программная инженерия. Методологические основы, Учебник. - М.: ТЕИС, 2006. - 608с.

8. Перечень ресурсов информационно-телекоммуникационной сети «Интернет»

1. Научная электронная библиотека eLIBRARY.RU – крупнейший российский информационный портал в области науки, технологии, медицины и образования, содержащий рефераты и полные тексты : [Электронный ресурс]. - Режим доступа URL:www.elibrary.ru.
2. Российское образование. Федеральный портал: [Электронный ресурс]. - Режим доступа URL: <http://www.edu.ru>.
3. Software Engineering Body of Knowledge (SWEBOOK) [Электронный ресурс] // The IEEE Computer Society. – USA, Washington, (2001-) – Режим доступа URL: <https://www.computer.org/education/bodies-of-knowledge/software-engineering> (дата обращения: 15.01.2019).
4. Стандарты и регламенты [Электронный ресурс] // Федеральное агентство по техническому регулированию и метрологии РФ – М.: Режим доступа URL: <https://www.gost.ru/portal/gost/home/standarts> (дата обращения: 15.01.2019).
5. Электронный фонд правовой и нормативно-технической документации [Электронный ресурс] // М.: АО «Кодекс». – Режим доступа URL: <http://docs.cntd.ru/> (дата обращения: 15.01.2019).
6. Microsoft Docs - Документация Майкрософт для пользователей, разработчиков и ИТ-специалистов – [Электронный ресурс] // Microsoft, М.: (2006-). – Режим доступа URL: <https://docs.microsoft.com/ru-ru/> (дата обращения: 15.01.2019).
7. Документации, обзоры, публикации по продуктам – [Электронный ресурс] // Компания «Интерфейс», М.: (2006-). – Режим доступа URL: <http://www.interface.ru/iservices/catalog.asp?catId=160> (дата обращения: 15.01.2019).

8. Липаев В.В. Программная инженерия: Комплекс учебников и монографий. – [Электронный ресурс] // Виртуальный компьютерный музей, М.: (2010-). – Режим доступа URL: <http://www.computer-museum.ru/books/lipaev/> (дата обращения: 15.01.2019).
9. Курсы для приобретения знаний и навыков, необходимых для успешной современной карьеры. // Oracle – Режим доступа URL: <https://academy.oracle.com/ru/solutions-curriculum-full.html> (дата обращения: 15.01.2019).
10. Методы и средства инженерии программного обеспечения: курс [Электронный ресурс] // сост. Лаврищева Е., Петрухин В. – М.: НОУ ИНТУИТ, 2011. Режим доступа URL: <https://www.intuit.ru/studies/courses/2190/237/info> (дата обращения: 15.01.2019).
11. Основы тестирования программного обеспечения: курс [Электронный ресурс] // сост. Котляров В. – М.: НОУ ИНТУИТ, 2011. Режим доступа URL: <https://www.intuit.ru/studies/courses/48/48/info> (дата обращения: 15.01.2019).
12. Верификация программного обеспечения: курс [Электронный ресурс] // сост. Налютин Н.Ю., Сеницын С.В. – М.: НОУ ИНТУИТ, 2011. Режим доступа URL: <https://www.intuit.ru/studies/courses/1040/209/info> (дата обращения: 15.01.2019).
13. Стандартизация и сертификация программного обеспечения: курс [Электронный ресурс] // сост. Позднеев Б. – М.: НОУ ИНТУИТ, 2011. Режим доступа URL: <https://www.intuit.ru/studies/courses/506/362/info> (дата обращения: 15.01.2019).
14. Основы объектно-ориентированного проектирования: курс [Электронный ресурс] // сост. Meyer, Bertrand. – М.: НОУ ИНТУИТ, 2011. Режим доступа URL: <https://www.intuit.ru/studies/courses/72/72/info> (дата обращения: 15.01.2019).
15. Объектно-ориентированное программирование и программная инженерия: курс [Электронный ресурс] // сост. Meyer, Bertrand. – М.: НОУ ИНТУИТ, 2011. Режим доступа URL: <https://www.intuit.ru/studies/courses/2309/609/info> (дата обращения: 15.01.2019).
16. Основы менеджмента программных проектов: курс [Электронный ресурс] // сост. Скопин И. – М.: НОУ ИНТУИТ, 2011. Режим доступа URL: <https://www.intuit.ru/studies/courses/38/38/info> (дата обращения: 15.01.2019).
17. Архитектурное проектирование программного обеспечения : курс [Электронный ресурс] // сост. Никитин И., Цулая М. – М.: НОУ ИНТУИТ, 2011. Режим доступа URL: <https://www.intuit.ru/studies/courses/3509/751/info> (дата обращения: 15.01.2019).

9 Методические указания для обучающихся по освоению дисциплины

Методика преподавания данной дисциплины предполагает чтение лекций, проведение практических занятий, групповых и индивидуальных консультаций по отдельным вопросам дисциплины. Предусмотрена самостоятельная работа студентов, а также прохождение аттестационных испытаний промежуточной аттестации (экзамен).

Лекции посвящаются рассмотрению наиболее важных концептуальных вопросов: основным вопросам программной инженерии, инструментам и методам программной инженерии; моделям и процессам управления проектом по разработке программных средств, навыками составления технической документации на различных этапах жизненного цикла информационной системы; разработки программного обеспечения, инструментами и методами программной инженерии, инсталляции программного и аппаратного обеспечения информационных и автоматизированных систем.

Целью проведения практических занятий является закрепление знаний студентов, полученных ими в ходе изучения дисциплины на лекциях и самостоятельно. Во время практических занятий выполняются лабораторные работы; на них разбираются конкретные ситуации по изучаемой теме. Для подготовки к занятиям практического типа студенты выполняют проработку рабочей программы, уделяя особое внимание целям и задачам, структуре и содержанию дисциплины, конспектирование источников и работу с конспектом лекций, просмотр рекомендуемой литературы.

При изучении дисциплины используются интерактивные методы обучения, такие

как:

1. Лекция:

– лекция-визуализация — подача материала осуществляется средствами технических средств обучения с кратким комментированием демонстрируемых визуальных материалов (презентаций).

2. Практические занятия:

– лабораторные работы — это вид учебной работы, в рамках которого осуществляется тот или иной эксперимент, направленный на получение результатов, имеющих значение с точки зрения успешного освоения студентами учебной программы.

Методические указания для курсового проектирования по дисциплине «Программная инженерия» для студентов направлений подготовки 09.03.03 «Прикладная информатика» / Рычка И.А., Чебанюк С.В.. – Петропавловск-Камчатский: КамчатГТУ, 2017. – 39 с.

10 Курсовой проект (работа)

10.1 Структура и содержание курсовой работы

Учебным планом предусмотрено выполнение курсовой работы по дисциплине. Курсовая работа имеет своей целью систематизацию теоретических знаний и практических навыков, полученных при изучении дисциплины, полученные студентами при изучении основного курса по дисциплине «Программная инженерия», выполнение обучающимся научного исследования конкретной проблемы. Курсовая работа представляет собой важный этап в подготовке к написанию выпускной квалификационной работы.

Курсовая работа включает в себя следующие элементы:

- титульный лист;
- бланк рецензии (рецензия на курсовую работу в общую нумерацию страниц не включается);
- задание;
- содержание;
- введение;
- основная часть;
- заключение;
- список использованных источников;
- приложения.

«Задание» включает в себя формулировку задания на выполнение курсовой работы и может быть уточнено в процессе выполнения работы.

«Содержание» (оглавление) – представляет собой структуру курсовой работы в виде перечня пунктов (заголовков) с указанием для каждого пункта номера страницы, с которой он начинается.

«Введение» может состоять из краткого описания области исследования, поставленной задачи (предмета курсового исследования) по теме выполняемой работы, формулировки цели курсовой работы и задач, решение которых позволит достичь цель работы, выбранных способов решения этих задач – инструментария исследования (методики, технологии, класс программного обеспечения), основных результатов анализа информации, необходимой для реализации задач курсовой работы, и самих результатов курсового проектирования (программный продукт, пояснительная записка, презентация). Формулировка цели курсовой работы совпадает с формулировкой темы. Перечень задач приводится в последовательности их решения.

Разделы основной части курсовой работы должны содержать описание результата

исследования проблемной области и объекта автоматизации (аналитический раздел), результат проектирования архитектуры, пользовательского интерфейса и детального проектирования программного средства (проектный раздел), конструирования и испытания проектных решений (раздел реализации и тестирования).

В заключении курсовой работы суммируются все выводы, подученные ее автором. В относительно небольшом объеме (2-3 страницы) студент в сжатой, лаконичной форме излагает наиболее важные, с его точки зрения, положения рассмотренной задачи, особо подчеркивая самостоятельность сделанных выводов. Самостоятельность и обоснованность выводов наиболее всего ценны в курсовой работе и существенно влияют на ее оценку.

В приложениях помимо обязательного составленного технического задания на разработку программного средства, могут быть включены руководства пользователя, программиста, коды программ с необходимыми комментариями, таблицы, рисунки и т.п., а также тексты, которые ввиду их громоздкости, большого количества или по другим причинам нецелесообразно размещать в других структурных элементах текстовой части курсовой работы.

10.2 Примерная тематика курсовых работ

1. Разработка программного обеспечения для автоматизированного учета движения основных средств.
2. Разработка программного обеспечения автоматизированной системы «Конкурс» для секретаря конкурсной комиссии организации.
3. Разработка программного обеспечения автоматизированной системы для калькуляции блюд в кафе.
4. Разработка программного обеспечения для автоматизированной системы учета приема заявок на объявление в ежедневной газете.
5. Разработка программного комплекса системы «Табель рабочего времени».
6. Разработка конфигурации 1С «Секретарь руководителя».
7. Разработка программного комплекса системы «Питомник».
8. Разработка программного комплекса автоматизированной системы «Спортивные соревнования».
9. Разработка программного комплекса автоматизированной системы «Учет материальных ценностей».
10. Разработка программного комплекса автоматизированной системы «Ревизия».
11. Разработка программного комплекса автоматизированной системы «Склад».
12. Разработка программного комплекса «Опрос потребителей».
13. Разработка программного комплекса «Ведение проекта».

Студент может предложить свою тему и после согласования с ведущим преподавателем выполнить курсовую работу на интересующую студента тему.

11 Перечень информационных технологий, используемых при осуществлении образовательного процесса по дисциплине, включая перечень программного обеспечения и информационно-справочных систем

11.1 Перечень информационных технологий, используемых при осуществлении образовательного процесса

- электронные образовательные ресурсы, представленные в п. 8 рабочей программы;
- использование слайд-презентаций;

- изучение межгосударственных стандартов ЕСПД на официальном сайте Росстандарта;
- интерактивное общение с обучающимися и консультирование посредством электронной почты.

11.2 Перечень программного обеспечения, используемого при осуществлении образовательного процесса

При освоении дисциплины используется лицензионное программное обеспечение:

- пакет Microsoft Office;
- Microsoft Visio;
- Microsoft Visual Studio;
- Microsoft Project;
- Java.

11.3 Перечень информационно-справочных систем

- справочная правовая система Консультант-плюс <http://www.consultant.ru/online>
- справочно-правовая система по законодательству Российской Федерации Гарант <http://www.garant.ru/online>

12 Материально-техническое обеспечение дисциплины

Для проведения занятий лекционного типа, групповых и индивидуальных консультаций, текущего контроля и промежуточной аттестации используются учебные аудитории 7-206, 7-401, 7-402, 7-501 с комплектом учебной мебели.

Для самостоятельной работы обучающихся, в том числе для курсового проектирования, используются кабинеты 7-401, 7-402, 7-501, 7-520; каждый кабинет оборудован:

- комплектом учебной мебели,
- компьютерами с доступом в информационно-телекоммуникационную сеть «Интернет» и в электронную информационно-образовательную среду организации,
- техническими средствами обучения для представления учебной информации: аудиторная доска, мультимедийное оборудование (ноутбук, проектор), наглядными пособиями.