

Федеральное государственное бюджетное образовательное учреждение
высшего образования
«Новосибирский государственный технический университет»

Кафедра теоретической и прикладной информатики

“УТВЕРЖДАЮ”
ДЕКАН ФПМИ

д.т.н. Тимофеев В. С.

“ ____ ” _____ г.

РАБОЧАЯ ПРОГРАММА УЧЕБНОЙ ДИСЦИПЛИНЫ
Программная инженерия

Образовательная программа: 02.04.03 Математическое обеспечение и администрирование информационных систем, магистерская программа: Математическое обеспечение информационных систем в экономике

Курс: 1, семестр : 1

Факультет прикладной математики и информатики,

		Семестр
№	Вид деятельности	1
1	Всего зачетных единиц (кредитов)	3
2	Всего часов	108
3	Всего занятий в контактной форме, час.	68
4	Лекции, час.	18
5	Практические занятия, час.	0
6	Лабораторные занятия, час.	36
7	из них в активной и интерактивной форме, час.	12
8	Аттестация, час.	2
9	Консультации, час.	12
10	Самостоятельная работа, час.	40
11	Виды самостоятельной работы (курсовой проект, курсовая работа, РГЗ, подготовка к контрольной работе)	
12	Вид аттестации	Э

Рабочая программа составлена на основании федерального государственного образовательного стандарта высшего образования по направлению (специальности): 02.04.03 Математическое обеспечение и администрирование информационных систем

ФГОС введен в действие приказом №1416 от 30.10.2014 г. , дата утверждения: 26.11.2014 г.

Место дисциплины в структуре учебного плана: Блок 1, базовая

Рабочая программа разработана на основе компетентностной модели выпускника по направлению (специальности): 02.04.03 Математическое обеспечение и администрирование информационных систем

Рабочая программа обсуждена на заседании кафедры

ТПИ, протокол заседания кафедры №4 от 20.06.2017

Утверждена на совете факультета прикладной математики и информатики, протокол № 6 от 21.06.2017

Программу разработал:

доцент, к.ф-м.н. Зайцев М. Г.

Заведующий кафедрой:

доцент, д.т.н. Чубич В. М.

Ответственный за образовательную программу:

декан Тимофеев В. С.

1. Внешние требования

Таблица 1.1

Компетенция ФГОС: ОПК.12 владение навыками выбора, проектирования, реализации, оценки качества и анализа эффективности программного обеспечения для решения задач в различных предметных областях; в части следующих результатов обучения:	
33. Знать модель FURPS+ требований к программному обеспечению	
35. Знать понятие жизненного цикла программного обеспечения и модели жизненного цикла	
у1. Уметь рассчитывать трудоёмкость программного обеспечения а основе диаграммы прецедентов и классов анализа, выполнять количественный анализ диаграмм UML.	
Компетенция ФГОС: ОПК.8 владение навыками использования современных системных программных средств: операционных систем, операционных и сетевых оболочек, сервисных программ; в части следующих результатов обучения:	
у1. Уметь использовать Visual Studio для выполнения модульных тестов, нагрузочных тестов и веб тестов производительности, тестов пользовательского интерфейса	
Компетенция ФГОС: ОПК.9 владение навыками разработки моделирующих алгоритмов и реализации их на базе языков и пакетов прикладных программ моделирования; в части следующих результатов обучения:	
у5. Уметь моделировать функциональные требования к программному обеспечению с помощью диаграмм прецедентов	
Компетенция ФГОС: ПК.4 готовность организовать работу в коллективе разработчиков программного обеспечения, на основе современных направлений развития методов и программных средств коллективной разработки программного обеспечения; в части следующих результатов обучения:	
31. Знать о проблемах, связанных с подбором и сохранением персонала в организации, занимающейся разработкой и сопровождением программного обеспечения	
32. Знать основные модели организации человеческой памяти, модели решения проблем и мотивации, а также механизмы их применения в практической работе руководителей проектами по созданию программного обеспечения.	

2. Требования НГТУ к результатам освоения дисциплины

Таблица 2.1

Результаты изучения дисциплины по уровням освоения (иметь представление, знать, уметь, владеть)	Формы организации занятий
<i>Программная инженерия</i>	
ПК.4.31 Знать о проблемах, связанных с подбором и сохранением персонала в организации, занимающейся разработкой и сопровождением программного обеспечения	
1. основные задачи, стоящие перед руководителем программного проекта	Лекции; Лабораторные работы; Самостоятельная работа
ПК.4.32 Знать основные модели организации человеческой памяти, модели решения проблем и мотивации, а также механизмы их применения в практической работе руководителей проектами по созданию программного обеспечения.	
2. основные модели организации человеческой памяти, модели решения проблем и мотивации, а также механизмы их применения в практической работе руководителей проектами по созданию программного обеспечения	Лекции; Лабораторные работы; Самостоятельная работа
ОПК.9.у5 Уметь моделировать функциональные требования к программному обеспечению с помощью диаграмм прецедентов	
3. описывать системные требования на структурированном естественном языке	Лекции; Самостоятельная работа
ОПК.12.33 Знать модель FURPS+ требований к программному обеспечению	
4. классификацию требований, которая получила название модели FURPS+	Лекции
ОПК.12.35 Знать понятие жизненного цикла программного обеспечения и модели жизненного цикла	

5.основные концепции, лежащие в основе процесса создания ПО и моделей этого процесса	Лекции
ОПК.12.у1 Уметь рассчитывать трудоёмкость программного обеспечения а основе диаграммы прецедентов и классов анализа, выполнять количественный анализ диаграмм UML.	
6.расчитывать трудоёмкость разработки ПО	Лекции
ОПК.8.у1 Уметь использовать Visual Studio для выполнения модульных тестов, нагрузочных тестов и веб тестов производительности, тестов пользовательского интерфейса	
7.Уметь использовать Visual Studio для выполнения модульных тестов, нагрузочных тестов и веб тестов производительности, тестов пользовательского интерфейса	Лекции

3. Содержание и структура учебной дисциплины

Таблица 3.1

Темы лекций	Активные формы, час.	Часы	Ссылки на результаты обучения	Учебная деятельность
Семестр: 1				
Дидактическая единица: Инженерия программного обеспечения				
1. Системотехника вычислительных систем	0	1	1	понимать, почему программное обеспечение все шире применяется в разработках системотехники; знать основные интеграционные характеристики систем: безотказность, производительность, надежность и безопасность; понимать, почему в процессе разработки систем необходимо учитывать окружение, в котором будет работать система; иметь представление о процессах создания систем и системного обеспечения

2. Процесс создания ПО	0	1	5	знать основные концепции, лежащие в основе (процесса создания ПО и моделей этого процесса; иметь представление об основных моделях процесса создания ПО и понимать, когда какую из них использовать; знать схему построения моделей процесса формирования требований к ПО, его разработки тестирования и модернизации; иметь понятие о CASE-технологиях, предназначенных для поддержки процесса создания ПО
3. Управление проектами	0	1	6	понимать различия между управлением программными проектами и управлением инженерными проектами других типов; знать основные задачи, стоящие перед руководителем программного проекта; понимать значимость и роль этапа планирования проекта среди всех этапов процесса создания ПО; знать графические способы представления графиков работ; иметь представление о типах рисков, возникающих при реализации программных проектов, и процессы управления этими рисками.
Дидактическая единица: Требования				

4. Требования к ПО	0	1	3, 4	иметь понятие о концепциях пользовательских системных требований и знать, почему для записи этих требований используются разные способы; понимать различия между функциональными нефункциональными требованиями; освоить два метода описания системных требований: основанный на структурированном естественном языке и основанный на языке программирования; знать стандарты документирования требований к программному обеспечению.
5. Разработка требований	0	1	4	знать основные процессы разработки требований и отношения между ними; ознакомиться с методами формирования и анализа требований; понимать важность аттестации требований; знать, почему необходимо управление требованиями.
6. Спецификация требований	0	1	1, 4	Функциональные требования. Нефункциональные требования модель FURPS+ Методы выявления требований: собеседование (интервьюирование); анкетирование; моделирование и анализ бизнес-процессов ; сессии по выявлению требований (мозговой штурм); создание и демонстрация пользователям работающих прототипов приложений (для выявления замечаний и дополнительных требований).
Дидактическая единица: Проектирование				

7. Архитектурное проектирование	0	1	5	понимать, для чего необходимо архитектурное проектирование ПО; знать различные модели, используемые при документировании системной архитектуры; иметь представление о разных типах архитектур ПО: структурной модели системы, модели системной декомпозиции по управлению и модели модульной декомпозиции; знать модели проблемно-зависимых архитектур, которые используются как основа для архитектур специализированных программных систем и как эталон при сравнении различных архитектур
8. Паттерны проектирования	0	1	5	Паттерны, их классификация. Паттерны проектирования в нотации языка UML. Паттерн Фасад и его обозначение в нотации языка UML. Паттерн Наблюдатель и его обозначение в нотации языка UML.
9. Проектирование пользовательского интерфейса	0	1	1	знать основные принципы проектирования интерфейса пользователя; освоить пять разных стилей взаимодействия пользователя с программными системами; знать разные стили представления информации и то, в каких случаях целесообразно графическое представление данных; познакомиться с основными правилами проектирования средств поддержки пользователя, встроенных в программное обеспечение; иметь представление об основных показателях удобства использования систем.
Дидактическая единица: Верификация и аттестация				

10. Верификация и аттестация ПО	0	1	5	знать отличие между верификацией и аттестацией ПО; познакомиться с инспектированием программ - эффективным методом выявления ошибок в программах; понять, почему статический анализ программ является одним из основных методов верификации; познакомиться с разработкой программ методом "чистая комната" и знать, почему этот метод оказывается эффективным.
11. Тестирование ПО	0	1	5, 7	знать, какие методы тестирования используются для выявления программных ошибок; познакомиться с основными принципами тестирования интерфейсов; знать особенности покомпонентного тестирования и тестирования процесса сборки объектно-ориентированных систем; познакомиться с CASE-средствами, применяемыми для тестирования
12. Надёжность систем	0	1	1	знать четыре составляющие функциональной надежности: работоспособность, безотказность, безопасность и защищенность; понимать, что критическими являются такие системы, при неправильном функционировании которых могут возникать тяжелые человеческие или экономические потери; знать основные способы достижения функциональной надежности системы: сведение к минимуму ошибок во время разработки, выявление и устранение ошибок во время эксплуатации и уменьшение последствий неправильного функционирования системы
Дидактическая единица: Управление				

13. Совершенствование производства ПО	0	1	5	знать основные принципы совершенствования производства ПО, а также понимать, насколько важен этот процесс; знать факторы, влияющие на качество программного продукта и производительность разработчиков ПО; понимать модель оценки уровня развития производства, которую можно применять для оценки качества технологии производства в больших организациях, разрабатывающих ПО; понимать, почему совершенствование на основе модели оценки уровня развития производства может быть применимо к производству не всех типов программного обеспечения
14. Управление качеством	0	2	1	знать основные процессы управления качеством, а также основные составляющие процесса управления качеством, а именно: обеспечение качества, планирование качества и контроль качества; понимать важность применения стандартов в процессе управления качеством; иметь представление о метрических показателях ПО и о различиях между прогнозируемыми и контролирующими показателями; понимать необходимость систем измерения при оценке показателей качества и знать об ограничениях в процессе измерения показателей ПО.

15. Управление персоналом	0	1	2	знать основные модели организации человеческой памяти, модели решения проблем и мотивации, а также механизмы их применения в практической работе руководителей проектами по созданию программного обеспечения; иметь представление об основных проблемах командной работы (в частности, при подборе членов команды), о сплоченности команды, внутри командных взаимоотношениях и ее структуре; знать о проблемах, связанных с подбором и сохранением персонала в организации, занимающейся разработкой и сопровождением программного обеспечения; иметь понятие о модели Р-СММ, которая является основой для повышения производительности сотрудников организации, занимающейся созданием программного обеспечения
---------------------------	---	---	---	--

16. Отбор команды разработчиков	0	2	2	объяснять важность отбора разработчиков проекта, имеющих различные черты характера, с целью формирования команды разработчиков программного проекта; описывать модели различных свойств характера; обсуждать индивидуальные стили работы, а также то, каким образом они воздействуют на ход выполнения проекта; описывать фазы командной разработки; описывать способы реализации общения; описывать централизованное и нецентрализованное документирование проекта, а также эффекты, проявляющиеся при организации взаимодействия между разработчиками проекта; обсуждать стили лидерства, а также то, каким образом они сказываются на команде.
---------------------------------	---	---	---	--

Таблица 3.2

Темы лабораторных работ	Активные формы, час.	Часы	Ссылки на результаты обучения	Учебная деятельность
Семестр: 1				
Дидактическая единица: Требования				
1. Формирование требований к универсальному калькулятору	3	3	1, 2	Формирование требований к универсальному калькулятору и их моделирование с помощью диаграммы прецедентов
8. Формирование и анализ требований к приложению "Измерение характеристик внимания"	0	3		Анализ предметной области. Формирование требований и описание с помощью модели прецедентов UML
9. . Разработка сценариев прецедентов	0	3		Разработка, специфицирование прецедентов с помощью шаблона
Дидактическая единица: Проектирование				
2. Иерархия классов "Число"	3	3	1, 2	Проектирование, спецификация и реализация иерархии классов Число
3. Иерархия классов "Редактор"	3	3	1, 2	Проектирование, специфицирование, реализация иерархии классов Число

4. Абстрактный тип данных (ADT) "Память"	3	3	1, 2	Проектирование, спецификация, реализация абстракции данных память на одно число
5. Управление калькулятором	0	3		Проектирование, спецификация, реализация класса Управление калькулятором со стереотипом "Entity"
6. Абстрактный тип данных: Процессор	0	3		Проектирование, спецификация, реализация абстракции данных Процессор
7. Интерфейс универсального калькулятора	0	3		Проектирование, спецификация, реализация класса Интерфейс в технологии визуального программирования
11. Анализ вариантов использования приложения "Измерение характеристик внимания"	0	3		Сформировать практические навыки идентификации основных абстракций предметной области
Дидактическая единица: Управление				
10. Оценка трудоёмкости разработки ПО для приложения Измерение характеристик внимания"	0	3		Оценка трудоёмкости разработки ПО на основе диаграммы прецедентов и классов анализа
12. Количественный анализ диаграмм использования	0	3		Сформировать практические навыки оценки качества моделей UML

4. Самостоятельная работа обучающегося

№	Виды самостоятельной работы	Ссылки на результаты обучения	Часы на выполнение	Часы на консультации
Семестр: 1				
1	Подготовка к занятиям	1, 2, 3	30	6
Подготовка к лекциям и лабораторным работам: Зайцев М. Г. Программная инженерия [Электронный ресурс] : электронный учебно-методический комплекс / М. Г. Зайцев ; Новосиб. гос. техн. ун-т. - Новосибирск, [2017]. - Режим доступа: http://elibrary.nstu.ru/source?bib_id=vtls000234633 . - Загл. с экрана.				
2	Подготовка к аттестации	1, 2, 3	10	6
Подготовка к сдаче экзамена: Зайцев М. Г. Программная инженерия [Электронный ресурс] : электронный учебно-методический комплекс / М. Г. Зайцев ; Новосиб. гос. техн. ун-т. - Новосибирск, [2017]. - Режим доступа: http://elibrary.nstu.ru/source?bib_id=vtls000234633 . - Загл. с экрана.				

5. Технология обучения

Для организации и контроля самостоятельной работы обучающихся, а также проведения консультаций применяются информационно-коммуникационные технологии (табл. 5.1).

Таблица 5.1

Деятельность	Информационно-коммуникационные технологии
Информирование	e-mail; Портал НГТУ; ЭБС
Консультирование	e-mail

Контроль	Портал НГТУ
Размещение учебных материалов	Портал НГТУ

Таблица 5.2

Активные и интерактивные формы проведения занятий

№	Наименование активных форм	Коды формируемых компетенций
1	Метод проектов	ПК.4;
Формируемые умения: з1. Знать о проблемах, связанных с подбором и сохранением персонала в организации, занимающейся разработкой и сопровождением программного обеспечения		
Краткое описание применения: Лабораторные работы организованы в два проекта. Каждый проект разбит на несколько лабораторных работ. Каждый проект завершается созданием работающего приложения.		
Подробная информация об использовании технологии приводится в "Зайцев М. . Технология разработки программного обеспечения [Электронный ресурс] : конспект лекций / М. Г. Зайцев ; Новосиб. гос. техн. ун-т. - Новосибирск, [2009]. - Режим доступа: http://moodle.ciu.nstu.ru/ . - Загл. с экрана."		

6. Правила аттестации обучающихся по учебной дисциплине

Для аттестации обучающихся по дисциплине используется балльно-рейтинговая система (БРС), позволяющая выставять оценки по традиционной шкале и 15-уровневой ECTS. Краткая информация о БРС приведена в табл. 6.1.

Таблица 6.1

Оцениваемые виды деятельности обучающихся	Максимальный балл
Семестр: 1	
<i>Подготовка к занятиям:</i>	
Контролирующие материалы (тесты) приводятся в "Зайцев М. Г. Программная инженерия [Электронный ресурс] : электронный учебно-методический комплекс / М. Г. Зайцев ; Новосиб. гос. техн. ун-т. - Новосибирск, [2017]. - Режим доступа: http://elibrary.nstu.ru/source?bib_id=vtls000234633 . - Загл. с экрана."	
<i>Дополнительная учебная деятельность:</i>	
<i>Лекция:</i>	20
Контролирующие материалы приводятся в "Зайцев М. Г. Программная инженерия [Электронный ресурс] : электронный учебно-методический комплекс / М. Г. Зайцев ; Новосиб. гос. техн. ун-т. - Новосибирск, [2017]. - Режим доступа: http://elibrary.nstu.ru/source?bib_id=vtls000234633 . - Загл. с экрана."	
<i>Лабораторная:</i>	40
Контролирующие материалы приводятся в "Зайцев М. Г. Программная инженерия [Электронный ресурс] : электронный учебно-методический комплекс / М. Г. Зайцев ; Новосиб. гос. техн. ун-т. - Новосибирск, [2017]. - Режим доступа: http://elibrary.nstu.ru/source?bib_id=vtls000234633 . - Загл. с экрана."	
<i>Экзамен:</i>	40
Контролирующие материалы приводятся в "Зайцев М. Г. Программная инженерия [Электронный ресурс] : электронный учебно-методический комплекс / М. Г. Зайцев ; Новосиб. гос. техн. ун-т. - Новосибирск, [2017]. - Режим доступа: http://elibrary.nstu.ru/source?bib_id=vtls000234633 . - Загл. с экрана."	

В таблице 6.2 представлено соответствие форм контроля заявляемым требованиям к результатам освоения дисциплины.

Таблица 6.2

Коды компетенций ФГОС	Результаты обучения	Формы контроля	
		Защита ЛР	Экзамен
ОПК.12	з3. Знать модель FURPS+ требований к программному обеспечению		+
	з5. Знать понятие жизненного цикла программного обеспечения и модели жизненного цикла	+	

	у1. Уметь рассчитывать трудоёмкость программного обеспечения а основе диаграммы прецедентов и классов анализа, выполнять количественный анализ диаграмм UML.		+
ОПК.8	у1. Уметь использовать Visual Studio для выполнения модульных тестов, нагрузочных тестов и веб тестов производительности, тестов пользовательского интерфейса		+
ОПК.9	у5. Уметь моделировать функциональные требования к программному обеспечению с помощью диаграмм прецедентов		+
ПК.4	з1. Знать о проблемах, связанных с подбором и сохранением персонала в организации, занимающейся разработкой и сопровождением программного обеспечения	+	+
	з2. Знать основные модели организации человеческой памяти, модели решения проблем и мотивации, а также механизмы их применения в практической работе руководителей проектами по созданию программного обеспечения.	+	+

Фонд оценочных средств по дисциплине представлен в приложении № 1 к рабочей программе.

7. Литература

Основная литература

1. Романов Е. Л. Программная инженерия [Электронный ресурс] : электронный учебно-методический комплекс / Е. Л. Романов ; Новосиб. гос. техн. ун-т. - Новосибирск, [2015]. - Режим доступа: http://elibrary.nstu.ru/source?bib_id=vtls000220170. - Загл. с экрана.
2. Технология разработки программных систем. Ч. 1 : методические указания к выполнению лабораторных работ по специальности 010503 "Математическое обеспечение и администрирование информационных систем", 010501 "Прикладная математика и информатика", 080801 "Прикладная информатика" / Новосиб. гос. техн. ун-т ; [сост. М. Г. Зайцев]. - Новосибирск, 2011. - 49, [2] с. : ил., табл. - Режим доступа: http://elibrary.nstu.ru/source?bib_id=vtls000155920
3. Грекул В. И. Проектирование информационных систем : учебное пособие / В. И. Грекул, Г. Н. Денищенко, Н. Л. Коровкина. - М., 2008. - 299 с.
4. Вендров А. М. Проектирование программного обеспечения экономических информационных систем : учебник для вузов по специальностям "Прикладная информатика (по областям)" и "Прикладная математика и информатика" / А. М. Вендров. - М., 2006. - 543 с. : ил.

Дополнительная литература

1. Крылов Е. В. Техника разработки программ. В 2 кн.. Кн. 2 : [учебник для вузов по направлениям "Информатика и вычислительная техника" и "Техника и технологии"] / Е. В. Крылов, В. А. Острейковский, Н. Г. Типикин. - М., 2008. - 468, [1] с. : ил., табл.
2. Котляров В. П. Основы тестирования программного обеспечения : учебное пособие / В. П. Котляров, Т. В. Коликова. - М., 2006. - 285 с. : ил.
3. Лафоре Р. Объектно-ориентированное программирование в С++ / Р. Лафоре ; [пер. с англ. А. Кузнецова, М. Назарова, В. Шрага]. - СПб., 2007. - 923 с. : ил.
4. Вендров А. М. Практикум по проектированию программного обеспечения экономических информационных систем : учебное пособие по специальностям "Прикладная информатика в экономике", "Математическое обеспечение и администрирование информационных систем" / А. М. Вендров. - М., 2006. - 191 с. : ил.

5. Грекул В. И. Проектирование информационных систем. Курс лекций : учебное пособие для вузов по специальностям в области информационных технологий / ?с В. И. Грекул, Г. Н. Денищенко, Н. Л. Коровкина ; Интернет ун-т информ. технологий. - М., 2005. - 298, [1] с. : ил.
6. Буч Г. Язык UML : руководство пользователя / Г. Буч, Д. Рамбо, И. Якобсон ; [пер. с англ. Н. Мухина]. - М., 2007. - 493 с. : ил.
7. Крылов Е. В. Техника разработки программ. В 2 кн.. Кн. 2 : [учебник для вузов по направлениям "Информатика и вычислительная техника" и "Техника и технологии"] / Е. В. Крылов, В. А. Острейковский, Н. Г. Типикин. - М., 2008. - 468, [1] с. : ил., табл.

Интернет-ресурсы

1. ЭБС НГТУ : <http://elibrary.nstu.ru/>
2. ЭБС «Издательство Лань» : <https://e.lanbook.com/>
3. ЭБС IPRbooks : <http://www.iprbookshop.ru/>
4. ЭБС "Znanium.com" : <http://znanium.com/>
5. :

8. Методическое и программное обеспечение

8.1 Методическое обеспечение

1. Зайцев М. Г. Программная инженерия [Электронный ресурс] : электронный учебно-методический комплекс / М. Г. Зайцев ; Новосиб. гос. техн. ун-т. - Новосибирск, [2017]. - Режим доступа: http://elibrary.nstu.ru/source?bib_id=vtls000234633. - Загл. с экрана.
2. Технология разработки программных систем. Ч. 2 : методические указания к выполнению лабораторных работ для 5 курса ФПМИ дневного отделения / Новосиб. гос. техн. ун-т ; [сост. М. Г. Зайцев]. - Новосибирск, 2012. - 35, [3] с. : ил., табл.. - Режим доступа: http://elibrary.nstu.ru/source?bib_id=vtls000172414
3. Зайцев М. . Технология разработки программного обеспечения [Электронный ресурс] : конспект лекций / М. Г. Зайцев ; Новосиб. гос. техн. ун-т. - Новосибирск, [2009]. - Режим доступа: <http://moodle.ciu.nstu.ru/>. - Загл. с экрана.

8.2 Специализированное программное обеспечение

1 Visual Studio

9. Материально-техническое обеспечение

Презентационное оборудование

№	Наименование	Назначение
1	Презентационное оборудование (мультимедиа-проектор, экран, компьютер для управления)	Применяется для электронных презентаций лекционных и практических занятий; для электронных презентаций публичных выступлений студентов

Компьютерный класс

№	Наименование	Назначение
1	Компьютерный класс (Компьютеры объединены в локальную сеть с выходом в Internet)	Компьютеры объединены в локальную сеть с выходом в Internet

1. Обобщенная структура фонда оценочных средств учебной дисциплины

Обобщенная структура фонда оценочных средств по дисциплине Программная инженерия приведена в Таблице.

Таблица

Формируемые компетенции	Показатели сформированности компетенций (знания, умения, навыки)	Темы	Этапы оценки компетенций	
			Мероприятия текущего контроля (курсовой проект, РГЗ(Р) и др.)	Промежуточная аттестация (экзамен, зачет)
ОПК.12 владение навыками выбора, проектирования, реализации, оценки качества и анализа эффективности программного обеспечения для решения задач в различных предметных областях	з3. Знать модель FURPS+ требований к программному обеспечению	Требования к ПО		Экзамен, вопросы 1-5
ОПК.12	з5. Знать понятие жизненного цикла программного обеспечения и модели жизненного цикла	Архитектурное проектирование		Экзамен, вопросы 6-10
ОПК.12	у1. Уметь рассчитывать трудоёмкость программного обеспечения а основе диаграммы прецедентов и классов анализа, выполнять количественный анализ диаграмм UML.	Управление проектами		Экзамен, вопросы 11-15
ОПК.8 владение навыками использования современных системных программных средств: операционных систем, операционных и сетевых оболочек, сервисных программ	у1. Уметь использовать Visual Studio для выполнения модульных тестов, нагрузочных тестов и веб тестов производительности , тестов пользовательского интерфейса	Тестирование ПО		Экзамен, вопросы 16-20
ОПК.9 владение навыками разработки моделирующих алгоритмов и реализации их на базе языков и пакетов прикладных	у5. Уметь моделировать функциональные требования к программному обеспечению с помощью диаграмм прецедентов	Требования к ПО		Экзамен, вопросы 21-27

программ моделирования				
ПК.4/ОУ готовность организовать работу в коллективе разработчиков программного обеспечения, на основе современных направлений развития методов и программных средств коллективной разработки программного обеспечения	31. Знать о проблемах, связанных с подбором и сохранением персонала в организации, занимающейся разработкой и сопровождением программного обеспечения	Системотехника вычислительных систем Формирование требований к универсальному калькулятору		Экзамен, вопросы 28-32
ПК.4/ОУ	32. Знать основные модели организации человеческой памяти, модели решения проблем и мотивации, а также механизмы их применения в практической работе руководителей проектами по созданию программного обеспечения.	Отбор команды разработчиков Управление персоналом		Экзамен, вопросы 33-36

2. Методика оценки этапов формирования компетенций в рамках дисциплины.

Промежуточная аттестация по дисциплине проводится в 1 семестре - в форме экзамена, который направлен на оценку сформированности компетенций ОПК.12, ОПК.8, ОПК.9, ПК.4/ОУ.

Экзамен проводится в устной форме по билетам. Билет формируется по следующему правилу: каждый вопрос из диапазона вопросов (список вопросов приведен ниже), входит в один или несколько билетов. В ходе экзамена преподаватель вправе задавать студенту дополнительные вопросы из общего перечня.

Кроме того, сформированность компетенций проверяется при проведении мероприятий текущего контроля, указанных в таблице раздела 1.

Общие правила выставления оценки по дисциплине определяются балльно-рейтинговой системой, приведенной в рабочей программе дисциплины.

На основании приведенных далее критериев можно сделать общий вывод о сформированности компетенций ОПК.12, ОПК.8, ОПК.9, ПК.4/ОУ, за которые отвечает дисциплина, на разных уровнях.

Общая характеристика уровней освоения компетенций.

Ниже порогового. Уровень выполнения работ не отвечает большинству основных требований, теоретическое содержание курса освоено частично, пробелы могут носить существенный характер, необходимые практические навыки работы с освоенным материалом сформированы не достаточно, большинство предусмотренных программой обучения учебных заданий не выполнены или выполнены с существенными ошибками.

Пороговый. Уровень выполнения работ отвечает большинству основных требований,

теоретическое содержание курса освоено частично, но пробелы не носят существенного характера, необходимые практические навыки работы с освоенным материалом в основном сформированы, большинство предусмотренных программой обучения учебных заданий выполнено, некоторые виды заданий выполнены с ошибками.

Базовый. Уровень выполнения работ отвечает всем основным требованиям, теоретическое содержание курса освоено полностью, без пробелов, некоторые практические навыки работы с освоенным материалом сформированы недостаточно, все предусмотренные программой обучения учебные задания выполнены, качество выполнения ни одного из них не оценено минимальным числом баллов, некоторые из выполненных заданий, возможно, содержат ошибки.

Продвинутый. Уровень выполнения работ отвечает всем требованиям, теоретическое содержание курса освоено полностью, без пробелов, необходимые практические навыки работы с освоенным материалом сформированы, все предусмотренные программой обучения учебные задания выполнены, качество их выполнения оценено числом баллов, близким к максимальному.

Формулировки критериев сформированности компетенций являются ориентировочными и могут быть изменены разработчиком.

Федеральное государственное бюджетное образовательное учреждение
высшего образования
«Новосибирский государственный технический университет»
Кафедра теоретической и прикладной информатики

Паспорт экзамена

по дисциплине «Программная инженерия», 1 семестр

1. Методика оценки

Экзамен проводится в устной форме, по билетам. Билет формируется по следующему правилу: первый вопрос выбирается из диапазона вопросов 1,4,8,12 ... и т.д., второй вопрос из диапазона вопросов 2,5,9,13, (список вопросов приведен ниже). В ходе экзамена преподаватель вправе задавать студенту дополнительные вопросы из общего перечня (п. 4).

Форма экзаменационного билета

НОВОСИБИРСКИЙ ГОСУДАРСТВЕННЫЙ ТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ
Факультет ФПМИ

Билет № _____

к экзамену по дисциплине «Программная инженерия»

1. Основные определения. Системный подход к проектированию ПО
2. Общие принципы проектирования систем
3. Диаграммы вариантов использования

Утверждаю: зав. кафедрой _____ должность, ФИО
(подпись) (дата)

2. Критерии оценки

- Ответ на экзаменационный билет считается **неудовлетворительным**, если студент при ответе на вопросы не дает определений основных понятий, не способен показать причинно-следственные связи явлений, оценка составляет 0 баллов.
- Ответ на экзаменационный билет засчитывается на **пороговом** уровне, если студент при ответе на вопросы дает определение основных понятий, может показать причинно-следственные связи явлений, оценка составляет 20 баллов.
- Ответ на экзаменационный билет засчитывается на **базовом** уровне, если студент при ответе на вопросы формулирует основные понятия, законы, дает характеристику процессов, явлений, проводит анализ причин, условий, может представить качественные характеристики процессов,

оценка составляет 30 баллов.

- Ответ на экзаменационный билет засчитывается на **продвинутом** уровне, если студент при ответе на вопросы проводит сравнительный анализ подходов, проводит комплексный анализ, выявляет проблемы, предлагает механизмы решения, способен представить количественные характеристики определенных процессов, приводит конкретные примеры из практики, оценка составляет 40 баллов.

3. Шкала оценки

В общей оценке по дисциплине экзаменационные баллы учитываются в соответствии с правилами балльно-рейтинговой системы, приведенными в рабочей программе дисциплины (не более 40 баллов).

4. Вопросы к экзамену по дисциплине «Программная инженерия»

1. Основные определения. Системный подход к проектированию ПО
2. Общие принципы проектирования систем
3. Диаграммы вариантов использования
4. Основные особенности проектов современных систем ПО
5. Визуальное моделирование
6. Диаграммы последовательности
7. Нормативно-методическое обеспечение создания ПО
8. Объектно-ориентированные методы анализа и проектирования ПО. Проблемы, стимулировавшие развитие ООП
9. Диаграммы кооперации
10. Функциональные и не функциональные требования
11. Основные принципы построения объектной модели
12. Диаграммы компонентов
13. Стандарт жизненного цикла ПО
14. Основные процессы ЖЦ ПО
15. Диаграммы классов
16. Вспомогательные процессы жизненного цикла ПО
17. Основные элементы объектной модели
18. Диаграммы состояний
19. Организационные процессы жизненного цикла ПО
20. Модели жизненного цикла ПО
21. Диаграммы размещения
22. Каскадная модель ЖЦ
23. Спецификация требований к программному обеспечению
24. Абстрагирование – принцип построения объектной модели
25. Каскадная модель ЖЦ
26. Инкапсуляция – принцип построения объектной модели
27. Элемент объектной модели – связь агрегации
28. Итерационная модель жизненного цикла
28. Модульность – принцип построения объектной модели
30. Элемент объектной модели – связь композиция
31. Образцы (паттерны)
32. Объектно-ориентированный анализ
33. Элемент объектной модели – связь обобщение

- 34. Что такое программное обеспечение
- 35. Оценка трудоемкости проекта по методике Rational Rose
- 36. Спецификация требований. Модель FURPS+

Вопросы к экзамену

1. Основные определения. Системный подход к проектированию ПО
 2. Общие принципы проектирования систем
 3. Диаграммы вариантов использования
 4. Основные особенности проектов современных систем ПО
 5. Визуальное моделирование
 6. Диаграммы последовательности
 7. Нормативно-методическое обеспечение создания ПО
 8. Объектно-ориентированные методы анализа и проектирования ПО.
- Проблемы, стимулировавшие развитие ООП
9. Диаграммы кооперации
 10. Функциональные и не функциональные требования
 11. Основные принципы построения объектной модели
 12. Диаграммы компонентов
 13. Стандарт жизненного цикла ПО
 14. Основные процессы ЖЦ ПО
 15. Диаграммы классов
 16. Вспомогательные процессы жизненного цикла ПО
 17. Основные элементы объектной модели
 18. Диаграммы состояний
 19. Организационные процессы жизненного цикла ПО
 20. Модели жизненного цикла ПО
 21. Диаграммы размещения
 22. Каскадная модель ЖЦ
 23. Спецификация требований к программному обеспечению
 24. Абстрагирование – принцип построения объектной модели
 25. Каскадная модель ЖЦ
 26. Инкапсуляция – принцип построения объектной модели
 27. Элемент объектной модели – связь агрегации
 28. Итерационная модель жизненного цикла
 28. Модульность – принцип построения объектной модели
 30. Элемент объектной модели – связь композиция
 31. Образцы (паттерны)
 32. Объектно-ориентированный анализ
 33. Элемент объектной модели – связь обобщение
 34. Что такое программное обеспечение
 35. Оценка трудоемкости проекта по методике Rational Rose
 36. Спецификация требований. Модель FURPS+

Обобщенная структура фонда оценочных средств учебной дисциплины

Тема	Код формируемой компетенции	Знания/умения	Контролирующее мероприятие (экзамен, зачет, курсовой проект и т.п.)
Архитектурное проектирование	ОПК.12	з5. Знать понятие жизненного цикла программного обеспечения и модели жизненного цикла	Экзамен – вопросы 1-14
Управление проектами		у4. Уметь рассчитывать трудоёмкость программного обеспечения а основе диаграммы прецедентов и классов анализа, выполнять количественный анализ диаграмм UML.	Лабораторная
Требования к ПО	ОПК.12 ОПК.9	з4. Знать модель FURPS+ требований к программному обеспечению у5. Уметь моделировать функциональные требования к программному обеспечению с помощью диаграмм прецедентов	Экзамен – вопросы 15-28
Формирование требований к универсальному калькулятору	ПК.4/ОУ	з1. Знать о проблемах, связанных с подбором и сохранением персонала в организации, занимающейся разработкой и сопровождением программного обеспечения	Лабораторная
Системотехника вычислительных систем		з1. Знать о проблемах, связанных с подбором и сохранением персонала в организации, занимающейся разработкой и сопровождением программного обеспечения	Экзамен – вопросы 29-42
Управление персоналом		з2. Знать основные модели организации человеческой памяти, модели решения проблем и мотивации, а также механизмы их применения в практической работе руководителей проектами по созданию программного обеспечения.	Экзамен – вопросы 43-56
Отбор команды разработчиков		з2. Знать основные модели организации человеческой памяти, модели решения проблем и мотивации, а также механизмы их применения в практической работе руководителей проектами по созданию программного обеспечения.	Лабораторная

Форма экзаменационного билета

Дисциплина Программная инженерия
(наименование дисциплины)

ЭКЗАМЕНАЦИОННЫЙ БИЛЕТ № 1

1 Жизненный цикл программного обеспечения

2 Пределы мышления

3 Модульная декомпозиция

.....

Составитель _____ И.О.Фамилия Зайцев М.Г
(подпись)

Заведующий кафедрой

_____ И.О.Фамилия Чубич В.М.
(подпись)

« ____ » _____ 2015 г.

Примечание * Структура экзаменационного билета утверждается на заседании кафедры. Практическая (ое) задача/задание может включаться по усмотрению преподавателя.

Критерии оценки

- Ответ засчитывается на **пороговом** уровне, если студент даёт исчерпывающий ответ на первый вопрос и по второму знает закон Миллера, оценка составляет 50 баллов
- Ответ засчитывается на **базовом** уровне, если студент даёт исчерпывающий ответ на первый и второй вопросы, оценка составляет 75 баллов
- Ответ засчитывается на **продвинутом** уровне, если студент даёт исчерпывающие ответы на все вопросы, оценка составляет 100 баллов

Экзамен считается сданным, если средняя сумма баллов по всем вопросам составляет не менее 50 баллов (по 100 балльной шкале).

Коэффициент, с которым учитывается полученная сумма баллов в общей оценке по дисциплине, определяется Правилами аттестации.

К паспорту экзаменационного билета прилагается полный перечень вопросов.