



Федеральное государственное бюджетное образовательное учреждение высшего образования
«ТЕХНОЛОГИЧЕСКИЙ УНИВЕРСИТЕТ ИМЕНИ ДВАЖДЫ ГЕРОЯ
СОВЕТСКОГО СОЮЗА, ЛЕТЧИКА-КОСМОНАВТА А.А. ЛЕОНОВА»

«УТВЕРЖДАЮ»

и.о. проректора

А.В. Троицкий

«__» _____ 2023 г.

**ИНСТИТУТ ИНФОКОММУНИКАЦИОННЫХ СИСТЕМ И
ТЕХНОЛОГИЙ**

**КАФЕДРА ИНФОРМАЦИОННЫХ ТЕХНОЛОГИЙ И УПРАВЛЯЮЩИХ
СИСТЕМ**

РАБОЧАЯ ПРОГРАММА

ДИСЦИПЛИНЫ

«ВВЕДЕНИЕ В ПРОГРАММНУЮ ИНЖЕНЕРИЮ»

Направление подготовки: 09.03.03 Прикладная информатика

Профиль: Прикладная информатика в системах управления

Уровень высшего образования: бакалавриат

Форма обучения: очная

Королев
2023

Рабочая программа является составной частью основной профессиональной образовательной программы и проходит рецензирование со стороны работодателей в составе основной профессиональной образовательной программы. Рабочая программа актуализируется и корректируется ежегодно.


Автор: Баранова О.М. Рабочая программа дисциплины: «Введение в программную инженерию». – Королев МО: ТУ, 2023 г.

Рецензент: к.т.н., доцент Исаева Г.Н.

Рабочая программа составлена в соответствии с требованиями федерального государственного образовательного стандарта высшего образования (ФГОС ВО) по направлению подготовки бакалавров 09.03.03 Прикладная информатика Учебного плана, утвержденного Ученым советом «ТУ».

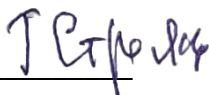
Протокол № 9 от 11.04.2022 г.

Рабочая программа рассмотрена и одобрена на заседании кафедры:

Заведующий кафедрой (ФИО, ученая степень, звание, подпись)	Артюшенко В.М. д.т.н. профессор 			
Год утверждения (переутверждения)	2023			
Номер и дата протокола заседания кафедры	№12 от 05.04.2023			

Рабочая программа согласована:

Руководитель ОПОП _____



Г.А. Стрельцова, к.т.н., доц.

Рабочая программа рекомендована на заседании УМС:

Год утверждения (переутверждения)	2023			
Номер и дата протокола заседания УМС	№ 5 от 11 апреля 2023 г.			

1. Перечень планируемых результатов обучения по дисциплине, соотнесенных с планируемыми результатами освоения ООП

Целью изучения дисциплины является:

- получение знаний и умений в области индустриальных способов (инженерных методов и средств) создания программного обеспечения (ПО);
- получение практических навыков работы с инструментальными средствами разработки программного обеспечения.

В процессе обучения студент приобретает и совершенствует следующие компетенции:

универсальные компетенции:

- способен определять круг задач в рамках поставленной цели и выбирать оптимальные способы их решения, исходя из действующих правовых норм, имеющихся ресурсов и ограничений (УК-2);

профессиональные компетенции:

- способен разрабатывать и адаптировать прикладное программное обеспечение (ПК-2).

Основными **задачами** дисциплины являются:

1. ознакомление с понятием качества программных продуктов, жизненным циклом программных продуктов и его моделями;
2. изучение методов анализа и проектирования ПО;
3. получение навыков применения современных CASE - средств поддержки этапов жизненного цикла ПО;
4. изучение методов верификации ПО;
5. приобретение навыков тестирования ПО в конкретной среде программирования.

Показатели освоения компетенций отражают следующие индикаторы:

Трудовые действия:

- В рамках поставленных задач определяет имеющиеся ресурсы и ограничения, действующие правовые нормы;
- Оценивает решение поставленных задач в зоне своей ответственности в соответствии с запланированными результатами контроля, при необходимости корректирует способы решения задач;
- Использует методы структурного и объектно-ориентированного программирования для разработки структуры программного кода ИС, верификации структуры программного кода ИС относительно архитектуры ИС и требований заказчика к ИС.

Необходимые умения:

- Определяет связи между поставленными задачами и ожидаемые результаты их решения;

- Анализирует план-график реализации проекта в целом и выбирает способ решения поставленных задач;
- Применяет языки программирования и работы с базами данных, инструменты и методы верификации структуры программного кода для разработки прототипа ИС в соответствии с требованиями.

Необходимые знания:

- Формулирует проблему, решение которой напрямую связано с достижением цели проекта;
- Анализирует языки программирования и работы с базами данных, инструменты и методы верификации структуры программного кода, современные объектно-ориентированные языки программирования, современные структурные языки программирования, языки современных бизнес-приложений для выполнения работ и управление работами по созданию (модификации) и сопровождению ИС.

2. Место дисциплины в структуре ООП ВО

Дисциплина «Введение в программную инженерию» относится к дисциплинам части, формируемой участниками образовательных отношений основной профессиональной образовательной программы подготовки бакалавров по направлению 09.03.03 «Прикладная информатика» и входит в состав модуля «Технологии программирования».

Изучение дисциплины базируется на знаниях, полученных в процессе изучения дисциплин «Теоретические основы информатики», «Введение в профессию», «Основы алгоритмизации и программирования» и компетенциях УК-2, ПК-2, УК-6, ПК-8, ОПК-2, ОПК-3, ОПК-7.

Знания и компетенции, полученные при освоении дисциплин модуля, являются базовыми при изучении дисциплин: «Объектно-ориентированное программирование», «Языки программирования», «Основы проектной деятельности (стандартизация документирования проекта» и при выполнении выпускной квалификационной работы бакалавра

3. Объем дисциплины и виды учебной работы

Общая трудоемкость модуля составляет 3 зачетных единицы, 108 часов.

Таблица 1

Виды занятий	Всего часов	Семестр 2	Семестр 4	Семестр 6	Семестр 7
Общая трудоемкость	108	108			
ОЧНАЯ ФОРМА ОБУЧЕНИЯ					
Аудиторные занятия	48	48			
Лекции (Л)	16	16			
Практические занятия (ПЗ)	32	32			
Лабораторные работы (ЛР)	-	-			
Самостоятельная работа	60	60			

Курсовые, расчетно-графические работы	–	–			
Контрольная работа, домашнее задание	+	+			
Текущий контроль знаний (7 - 8, 15 - 16 недели)	Тест	+			
Вид итогового контроля	экзамен	экзамен			

4. Содержание дисциплины (модуля)

4.1. Темы дисциплины и виды занятий

Таблица 2

Наименование тем	Лекции, час. Очн./Заочн.	Практические занятия, час Очн./Заочн	Занятия в интерактивной форме, час Очн./Заочн	Практическая подготовка, час	Код компетенций
Тема 1. Жизненный цикл ПО	2	2	2	2	УК-2
Тема 2. Методы анализа и проектирования ПО	4	8	2	2	УК-2
Тема 3. Язык UML	4	8	3	4	ПК-6
Тема 4. Качество, надежность и эффективность ПО	2	6	2	4	ПК-6
Тема 5. Интегрированные среды проектирования и разработки ПО	4	8	3	4	ПК-6
Итого:	16	32	12	16	

4.2. Содержание тем дисциплины

Тема 1. Жизненный цикл ПО

Системный подход к проектированию ПО. Понятие программного обеспечения, проекта и проектирования ПО. Программная инженерия. Основные требования, предъявляемые к программному изделию, как к продукту производственно-технического назначения.

Стандарты в области программного обеспечения. Сертификация и оценка процессов создания ПО. Определение жизненного цикла ПО. Стандарты, опре-

деляющие этапы и процессы жизненного цикла (ЖЦ). Модели и стадии ЖЦ ПО.

Тема 2. Методы анализа и проектирования ПО

Проблемы проектирования ПО. Структурные методы анализа и проектирования ПО. Метод функционального моделирования SADT (IDEF0). Метод моделирования процессов IDEF3. Моделирование потоков данных DFD. Моделирование данных.

Основные принципы и свойства объектно-ориентированного проектирования. Основные элементы объектной модели. Объекты, свойства объектов. Понятие классов. Общая характеристика классов. Пример использования объектно-ориентированного подхода. Сопоставление и взаимосвязь структурного и объектно-ориентированного подходов.

Тема 3. Язык UML

Язык UML: диаграммы и их назначение. Применение UML для выполнения этапов анализа и проектирования. Унифицированный процесс разработки ПО.

Тема 4. Качество, надежность и эффективность ПО

Понятие качества программного продукта. Показатели качества программного продукта средств. Пути повышения качества программных продуктов. Числовые характеристики объектно-ориентированных программ.

Основные понятия и показатели надежности программных средств. Классификация программных ошибок. Принципы и методы обеспечения надежности программных средств.

Отладка и тестирование ПО. Методы тестирования ПО. ГОСТ Р ИСО/МЭК 12119-2000. Работы по тестированию. Протоколы тестирования. Отчет о тестировании.

Понятие зрелости процессов создания ПО. Модель оценки зрелости CMM (стандарт ISO/IEC TR 15504-CMM). Управление требованиями при разработке ПО. Управление конфигурацией при разработке ПО.

Тема 5. Интегрированные среды проектирования и разработки ПО

Обзор сред проектирования и разработки ПО. Язык C# и особенности работы в среде Microsoft Visual Studio 2013, разработка интерфейса пользователя и многооконных приложений. Работа со стандартными классами. Язык LINQ и его использование для работы с коллекциями.

Эволюция методов доступа к удаленным данным. Технология ADO.NET для доступа к данным. Анализ технологий интегрированных программных сред для разработки приложений с БД.

5. Перечень учебно-методического обеспечения для самостоятельной работы по дисциплине

1. «Методические указания для обучающихся по освоению дисциплины «Введение в программную инженерию»

6. Фонд оценочных средств для проведения промежуточной аттестации обучающихся по дисциплине

Структура фонда оценочных средств приведена в Приложении 1.

7. Перечень основной и дополнительной учебной литературы, необходимой для освоения дисциплины

Основная литература:

1. В.А. Антипов, А.А. Бубнов, А.Н. Пылькин, В.К. Столчнев. Введение в программную инженерию: Учебник. М.: КУРС: ИНФРА-М, 2022. — 336 с. ЭБС Знаниум. Режим доступа: <http://znanium.com/bookread2.php?book=850951>
2. Гагарина Л.Г., Кокорева Е.В., Сидорова-Виснадул Б.Д. Технология разработки программного обеспечения : учеб. пособие / под ред. Л.Г. Гагариной. — М.: ИД «ФОРУМ» : ИНФРА-М, 2023. — 400 с. / ЭБС «Знаниум», режим доступа: <http://znanium.com/bookread2.php?book=924760>
3. Антамошкин О.А. Программная инженерия. Теория и практика. - 1. - Красноярск : Сибирский федеральный университет, 2021. - 247 с. - ISBN 9785763825114. - Электронная программа (визуальная). Электронные данные : электронные. URL: <http://znanium.com/go.php?id=492527>

Дополнительная литература:

1. Влацкая И.В., Заельская Е.А., Надточий Е.С. Проектирование и реализация прикладного программного обеспечения. – Оренбург: ОГУ. – 2023. – 119 с. ЭБС «Университетская библиотека онлайн». Режим доступа: http://biblioclub.ru/index.php?page=book_view_red&book_id=439107
2. Заботина Н.Н. Проектирование информационных систем : Учебное пособие. - 1. - Москва : ООО "Научно-издательский центр ИНФРА-М", 2021. - 331 с. - ISBN 9785160045092. - Электронная программа (визуальная). Электронные данные : электронные. URL: <http://znanium.com/go.php?id=1036508>

8. Перечень ресурсов информационно-телекоммуникационной сети «Интернет», необходимых для освоения дисциплины

Интернет-ресурсы:

1. <http://www.intuit.ru> - образовательный портал,
2. <http://www.infojournal.ru> - научно-образовательный портал,
3. <http://msdn.microsoft.com/ru-ru/library/> - портал разработчика Microsoft,
4. <http://www.interface.ru/> - научно-образовательный портал.

5. <http://www.biblioclub.ru> – электронная библиотечная система
6. <http://znanium.com> – электронная библиотечная система

9. Методические указания для обучающихся по освоению дисциплины

Методические указания для обучающихся по освоению дисциплины приведены в Приложении 2 к настоящей рабочей программе.

10. Перечень информационных технологий, используемых при осуществлении образовательного процесса по дисциплине

Перечень программного обеспечения: Microsoft Visio, Microsoft Visual Studio, Modelio Open Source.

Информационные справочные системы:

1. Ресурсы информационно-образовательной среды «ТУ».
2. Рабочая программа и методическое обеспечение по дисциплине «Введение в программную инженерию».

1. Описание материально-технической базы, необходимой для осуществления образовательного процесса по дисциплине

Лекционные занятия:

- аудитория, оснащенная презентационной техникой (проектор, экран);
- комплект электронных презентаций / слайдов;
- программные продукты Microsoft Visual Studio.

Прочее:

- рабочее место преподавателя, оснащенное компьютером с доступом в Интернет;
- рабочие места студентов, оснащенные компьютерами с доступом в Интернет.

Практические занятия:

- Аудитория, оснащенная мультимедийными средствами (интерактивная доска).
- рабочее место преподавателя, оснащенное ПК с доступом в глобальную сеть Интернет;
- рабочие места студентов, оснащенные компьютерами с доступом в глобальную сеть Интернет и установленным программным обеспечением.

**Фонд оценочных средств для проведения промежуточной аттестации
обучающихся по дисциплине**

**ИНСТИТУТ ИНФОКОММУНИКАЦИОННЫХ СИСТЕМ И
ТЕХНОЛОГИЙ**

**КАФЕДРА ИНФОРМАЦИОННЫХ ТЕХНОЛОГИЙ И УПРАВЛЯЮЩИХ
СИСТЕМ**

**ФОНД ОЦЕНОЧНЫХ СРЕДСТВ ДЛЯ ПРОВЕДЕНИЯ
ПРОМЕЖУТОЧНОЙ АТТЕСТАЦИИ ОБУЧАЮЩИХСЯ ПО
ДИСЦИПЛИНЕ**

«Введение в программную инженерию»

(Приложение 1 к рабочей программе)

Направление подготовки: 09.03.03 – Прикладная информатика

Профиль: Прикладная информатика в системах управления

Уровень высшего образования: бакалавриат

Форма обучения: очная

**Королев
2023**

1. Перечень компетенций с указанием этапов их формирования в процессе освоения образовательной программы

№ п/п	Индекс компетенции	Содержание компетенции (или ее части)*	Раздел дисциплины, обеспечивающий формирование компетенции (или ее части)	В результате изучения раздела дисциплины, обеспечивающего формирование компетенции (или ее части), обучающийся должен:		
				трудовые действия	необходимые умения	необходимые знания
1.	УК-2	Способен определять круг задач в рамках поставленной цели и выбирать оптимальные способы их решения, исходя из действующих правовых норм, имеющихся ресурсов и ограничений	Темы 1, 2	В рамках поставленных задач определяет имеющиеся ресурсы и ограничения, действующие правовые нормы Оценивает решение поставленных задач в зоне своей ответственности в соответствии с запланированными результатами контроля, при необходимости корректирует способы решения задач	Определяет связи между поставленными задачами и ожидаемые результаты их решения Анализирует план-график реализации проекта в целом и выбирает способ решения поставленных задач	Формулирует проблему, решение которой напрямую связано с достижением цели проекта
2.	ПК-2	Способен разрабатывать и адаптировать прикладное программное обеспечение;	Темы 3,4,5	Использует методы структурного и объектно-ориентированного программирования для разработки структуры программного кода ИС, верификации структуры программного кода ИС относительно архитектуры ИС и требований заказчика к ИС	Применяет языки программирования и работы с базами данных, инструменты и методы верификации структуры программного кода для разработки прототипа ИС в соответствии с требованиями	Анализирует языки программирования и работы с базами данных, инструменты и методы верификации структуры программного кода, современные объектно-ориентированные языки программирования, современные структурные языки программирования, языки современных бизнес-приложений для выполнения работ и управление работами по созданию (модификации) и сопровождению ИС

2. Описание показателей и критериев оценивания компетенций на различных этапах их формирования, описание шкал оценивания

Код компетенции	Инструменты, оценивающие сформированность компетенции	Показатель оценивания компетенции	Критерии оценки
УК-2, ПК-2	Практическое задание	<p>А) полностью сформирована (компетенция освоена на высоком уровне) – 5 баллов</p> <p>Б) частично сформирована:</p> <ul style="list-style-type: none"> •компетенция освоена на продвинутом уровне – 4 балла; •компетенция освоена на базовом уровне – 3 балла; <p>В) не сформирована (компетенция не освоена) – 2 и менее баллов</p>	<p>1. Проводится в форме письменной работы</p> <p>2.Время, отведенное на процедуру – 10 - 15 мин.</p> <p>Неявка – 0.</p> <p>Критерии оценки:</p> <p>1.Соответствие ответа заявленной тематике (0-5 баллов).</p> <p>Максимальная сумма баллов - 5 баллов.</p> <p>Результаты оценочной процедуры представляются обучающимся в срок не позднее 1 недели после проведения процедуры – для текущего контроля. Оценка проставляется в электронный журнал.</p>
УК-2	Доклад в форме презентации	<p>А) полностью сформирована (компетенция освоена на высоком уровне) – 5 баллов</p> <p>Б) частично сформирована:</p> <ul style="list-style-type: none"> •компетенция освоена на продвинутом уровне – 4 балла; •компетенция освоена на базовом уровне – 3 балла; <p>В) не сформирована (компетенция не освоена) – 2 и менее баллов</p>	<p>Проводится устно с использованием мультимедийных систем, а также с использованием технических средств</p> <p>Время, отведенное на процедуру – 10 - 15 мин.</p> <p>Неявка – 0.</p> <p>Критерии оценки:</p> <p>1.Соответствие представленной презентации заявленной тематике (1 балл).</p> <p>2.Качество источников и их количество при подготовке доклада и разработке презентации (1 балл).</p> <p>3.Владение информацией и способность отвечать на вопросы аудитории (1 балл).</p> <p>4.Качество самой представленной презентации (1 балл).</p> <p>5.Оригинальность подхода и всестороннее раскрытие выбранной тематике (1 балл).</p> <p>Максимальная сумма баллов - 5 баллов.</p> <p>Результаты оценочной процедуры представляются</p>

			обучающимся в срок не позднее 1 недели после проведения процедуры – для текущего контроля. Оценка проставляется в электронный журнал.
--	--	--	---------------------------------------------------------------------------------------------------------------------------------------

УК-2, ПК-2	Контрольная работа (проводится в качестве закрепления освоенного курса и компетенций) для студентов всех форм обучения	А)полностью сформирована – 5 баллов Б)частично сформирована – 3-4 балла В)не сформирована- менее 2 и менее баллов	Критерии оценки контрольной работы: 1.Соответствие содержания контрольной работы заявленной тематике (1 балл). 2.Качество источников и их количество при подготовке работы (1 балл). 3.Владение информацией и способность отвечать на вопросы аудитории (1 балл). 4.Качество самой представленной работы (1 балл). 5.Оригинальность подхода и всестороннее раскрытие выбранной тематики (1 балл). Максимальная сумма баллов - 5 баллов Оценка проставляется в журнал
------------	-------------------------------------------------------------------------------------------------------------------------------	-------------------------------------------------------------------------------------------------------------------------	--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

3. Типовые контрольные задания или иные материалы, необходимые для оценки знаний, умений, навыков и (или) опыта деятельности, характеризующих этапы формирования компетенций в процессе освоения образовательной программы

Тематика практического задания

1. Разработать IDEF0-модель для процесса записи к врачу и описать её.
2. Разработать DFD-модель для процесса записи к врачу и описать её.
3. Разработать IDEF0-модель для процесса записи в школу и описать её.
4. Разработать DFD-модель для процесса записи в школу и описать её.
5. Разработать IDEF0-модель для процесса бронирования номеров в гостинице и описать её.
6. Разработать DFD-модель для процесса бронирования номеров в гостинице и описать её.
7. Разработать IDEF0-модель для процесса заказа товаров и описать её.
8. Разработать DFD-модель для процесса заказа товаров и описать её.
9. Составить на языке UML диаграмму вариантов использования для программной системы удаленной записи к врачу и описать её.
10. Составить на языке UML диаграмму вариантов использования для программной системы удаленной записи в школу и описать её.
11. Составить на языке UML диаграмму вариантов использования для программной системы бронирования номеров в гостинице и описать её.

12. Составить на языке UML диаграмму вариантов использования для программной системы заказа товаров и описать её.
13. Составить на языке UML диаграмму деятельности для программной системы удаленной записи к врачу и описать её.
14. Составить на языке UML диаграмму деятельности для программной системы удаленной записи в школу и описать её
15. Составить на языке UML диаграмму деятельности для программной системы бронирования номеров в гостинице и описать её заказа товаров и описать её
16. Составить на языке UML диаграмму деятельности для программной системы
17. Описать классы предметной области и построить диаграмму классов UML для программной системы удаленной записи к врачу.
18. Описать классы предметной области и построить диаграмму классов UML для программной системы бронирования номеров в гостинице
19. Описать классы предметной области и построить диаграмму классов UML для программной системы заказа товаров
20. Описать классы предметной области и построить диаграмму классов UML для программной системы Call-центра.
21. Разработать тесты и провести тестирование заданного компонента методом «белого ящика».
22. Разработать тесты и провести тестирование заданного компонента методом «черного ящика».
23. Разработать тесты и провести функциональное тестирование заданного компонента.
24. Разработать прототип интерфейса пользователя на основе диаграммы вариантов использования для программной системы удаленной записи к врачу в заданной инструментальной среде программирования.
25. Разработать прототип интерфейса пользователя на основе диаграммы вариантов использования для программной системы удаленной записи в школу в заданной инструментальной среде программирования.

Тематика докладов

1. CASE-средства проектирования моделей ПО.
2. Инструментальные средства построения моделей ПО.
3. Технологии информационного моделирования.
4. Применение диаграмм UML для решения практической задачи.
5. Документирование диаграмм UML.
6. Процессы программного проекта
7. Методы определения программных требований
8. Программные средства тестирования ПО
9. Метрики и атрибуты качества ПО.
10. Методы отладки в различных программных средах.
11. Характеристика метода нисходящего тестирования.
12. Пошаговое и монолитное тестирование модулей.
13. Технология создание проектов с использованием среды Microsoft Visual Studio.
14. Назначение и описание класса

15. Сравнительный анализ сред программирования

4. Методические материалы, определяющие процедуры оценивания знаний, умений, навыков и (или) опыта деятельности, характеризующих этапы формирования компетенций

Формой контроля знаний по дисциплине «Введение в программную инженерию» являются две текущие аттестации в виде тестов и промежуточная аттестация в форме экзамена в устной форме.

Неделя текущего контроля	Вид оценочного средства	Код компетенций, оценивающий знания, умения, навыки	Содержание оценочного средства	Требования к выполнению	Срок сдачи (неделя семестра)	Критерии оценки по содержанию и качеству с указанием баллов
В соответствии с графиком учебного процесса	тестирование	УК-2, ПК-2	20 вопросов	Компьютерное тестирование; время, отведенное на процедуру - 30 минут	Результаты тестирования предоставляются в день проведения процедуры	Критерии оценки определяются процентным соотношением. Не явка - Удовлетворительно - от 51% правильных ответов. Хорошо - от 70%. Отлично – от 90%.
	тестирование	УК-2, ПК-2	20 вопросов	Компьютерное тестирование; время, отведенное на процедуру – 30 минут	Результаты тестирования предоставляются в день проведения процедуры	Критерии оценки определяются процентным соотношением. Не явка - 0 Удовлетворительно - от 51% правильных ответов. Хорошо - от 70%. Отлично – от 90%. Максимальная оценка – 5 баллов.
	Зачет с оценкой	УК-2, ПК-2	2 вопроса и задача	Зачет проводится в устной форме,	Результаты предоставляются	Критерии оценки: «Отлично»:

				<p>путем ответа на вопросы. Время, отведенное на процедуру – 30 минут.</p>	<p>в день проведения экзамена</p>	<ol style="list-style-type: none"> 1. знание основных понятий предмета; 2. умение использовать и применять полученные знания на практике; 3. работа на практических занятиях; 4. знание основных научных теорий, изучаемых предметов; 5. ответ на вопросы билета. <p>«Хорошо»:</p> <ul style="list-style-type: none"> • знание основных понятий предмета; • умение использовать и применять полученные знания на практике; • работа на практических занятиях; • знание основных научных теорий, изучаемых предметов; • ответы на вопросы билета • неправильно решено практическое задание <p>«Удовлетворительно»:</p> <ol style="list-style-type: none"> 1. демонстрирует ча-
--	--	--	--	----------------------------------------------------------------------------	-----------------------------------	---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

						<p>стичные знания по темам дисциплин;</p> <p>2. незнание неумение использовать и применять полученные знания на практике;</p> <p>3. не работал на практических занятиях;</p> <p>«Неудовлетворительно»:</p> <p>1. демонстрирует частичные знания по темам дисциплин;</p> <p>2. незнание основных понятий предмета;</p> <p>3. неумение использовать и применять полученные знания на практике;</p> <p>4. не работал на практических занятиях;</p> <p>5. не отвечает на вопросы.</p>
--	--	--	--	--	--	------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

* *Итоговое начисление баллов по дисциплине осуществляется в соответствии с разработанной и внедренной балльно-рейтинговой системой контроля и оценивания уровня знаний и внеучебной созидательной активности обучающихся, согласно приказу «О внедрении новой балльно-рейтинговой системы контроля и оценивания уровня знаний и внеучебной созидательной активности обучающихся»*

4.1. Тестовые задания

Тесты используются в режиме промежуточного контроля. По форме заданий выбраны закрытые тесты (с выборочным ответом).

Тема 1: Жизненный цикл ПО

1. Определите понятие жизненного цикла ПО
2. Модель жизненного цикла – это
3. Назовите модели жизненного цикла ПО
4. Укажите правильную последовательность этапов при каскадной модели жизненного цикла.
5. В чем достоинства и недостатки спиральной модели?
6. Определите специфику спиральной модели жизненного цикла
7. Назовите группы процессов жизненного цикла в соответствии со стандартом ГОСТ Р ИСО/МЭК 12207.

Тема 2. Методы анализа и проектирования ПО

8. Проектирование ПО в основном рассматривается как...
9. Что из приведенного не является одним из методов проектирования программного обеспечения?
10. Техническое задание — это
11. Архитектура программной системы — это
12. Как называется процесс разбиения одной сложной задачи на несколько простых подзадач?
13. Назначение интерфейс заключается в...?
14. При конструировании программного обеспечения на этапе разработки или выбора алгоритма решения реализуется следующее...
15. Какое свойство определяет процедуры внесения изменений в требования?
16. Что такое управление сборками?
17. Какова основная задача удовлетворения потребителя

Тема 3. Язык UML

18. UML — это ...
19. Определите назначение диаграммы прецедентов.
20. Назовите основные элементы диаграммы прецедентов
21. Для чего применяются варианты использования (Use Case)?
22. Кто может быть актером в Use Case-диаграммах?
23. Что является характеристикой точки зрения?
24. Какие диаграммы используются для моделирования аппаратной части системы, с которой связано ПО?

Тема 4. Качество, надежность и эффективность ПО

25. Что означает легкость применения программного обеспечения?
26. Мобильность программного обеспечения это:
27. Что определяет устойчивость программного обеспечения?
28. Определите задачи этапа тестирования
29. На этапе тестирования пользователь выполняет следующее:
30. Что из приведенного является критериями оценки удобства интерфейсов?
31. Что означает валидация ПО?
32. При выполнении какого вида тестирования две и более компонент тестируются на совместимость?
33. Что такое интеграционное тестирование?

Тема 5. Интегрированные среды проектирования и разработки ПО

34. Определите структуру проекта в Microsoft Visual Studio

35. Что означает значение public модификатора доступа?
36. Что означает значение модификатора доступа?
37. Для чего используются индексы?
38. Приведенное ниже описание задает...
39. Что реализует приведенный ниже запрос на LINQ
40. Назовите преимущества использования LINQ

4.3. Вопросы, выносимые на зачет

1. Подходы к разработке программных средств. Их краткая характеристика.
2. Жизненный цикл программного обеспечения. Основные понятия.
3. Модели жизненных циклов программного обеспечения, их характеристики и области применения.
4. Особенности модели жизненного цикла «спираль»
5. Содержание этапов анализа и проектирования при разработке программного обеспечения по методу функциональной декомпозиции.
6. Краткая характеристика объектно-ориентированного подхода к разработке программного обеспечения. Понятия «Класс» и «объект».
7. Виды контроля программ; статические и динамические методы контроля
8. Тестирование и отладка, подходы к тестированию.
9. Методы функционального тестирования.
10. Методы структурного тестирования. Тестирование путей, цикломатическая сложность программы.
11. Совместное использование функционального и структурного тестирования.
12. Тестирование программных комплексов, построенных методом функциональной декомпозиции
13. Тестирование программ и жизненный цикл программного продукта.
14. Общая характеристика и назначение языка UML.
15. Диаграммы вариантов использования, назначение, компоненты, отношения между компонентами.
16. Диаграмма последовательности и кооперативная диаграмма, их назначение, компоненты.
17. Диаграмма классов, характеристики класса
18. Диаграмма классов, типы и характеристики отношений.
19. Диаграммы состояний, их назначение, компоненты.
20. Диаграммы деятельности, их назначение, компоненты.
21. Диаграммы компонентов и размещения, их назначение, составные части.
22. Унифицированный процесс разработки программных средств. Общая характеристика и этапы.
23. основополагающие принципы разработки программных средств по унифицированному процессу.
24. Процесс определения требований к программным средствам. Документирование требований с помощью диаграмм.

25. Процесс анализа и структурирования требований при разработке программных средств.
26. Процесс проектирования при разработке программных средств. Основные отличия моделей анализа и проектирования.
27. Особенности тестирования программных средств, построенных по объектно-ориентированной методике. Тестирование классов.
28. Тестирование взаимодействия классов. Контрактное и защитное программирование. Тестирование иерархии классов.
29. Средства тестирования в Microsoft Visual Studio 2010.
30. Сравнение объектно-ориентированного и процедурного программирования.
31. Особенности языка программирования C# и среды Microsoft Visual Studio 2010.
32. Особенности работы с массивами на C#
33. Объявление классов на C#.
34. Особенности использования формальных и фактических параметров на C# (out, ref). Составление процедурных программ.
35. Принципы стыковки Visual Studio с базами данных

Методические указания для обучающихся по освоению дисциплины

**ИНСТИТУТ ИНФОКОММУНИКАЦИОННЫХ СИСТЕМ И
ТЕХНОЛОГИЙ**

**КАФЕДРА ИНФОРМАЦИОННЫХ ТЕХНОЛОГИЙ И УПРАВЛЯЮЩИХ
СИСТЕМ**

**МЕТОДИЧЕСКИЕ УКАЗАНИЯ ДЛЯ ОБУЧАЮЩИХСЯ ПО
ОСВОЕНИЮ ДИСЦИПЛИНЫ**

«Введение в программную инженерию»

(Приложение 2 к рабочей программе)

Направление подготовки: 09.03.03 – Прикладная информатика

Профиль: Прикладная информатика в системах управления

Уровень высшего образования: бакалавриат

Форма обучения: очная

**Королев
2023**

1. Общие положения

Целью изучения дисциплины является:

- получение знаний и умений в области индустриальных способов (инженерных методов и средств) создания программного обеспечения (ПО);
- получение практических навыков работы с инструментальными средствами разработки программного обеспечения.

Задачи дисциплины:

1. ознакомление с понятием качества программных продуктов, жизненным циклом программных продуктов и его моделями;
2. изучение методов анализа и проектирования ПО;
3. получение навыков применения современных CASE - средств поддержки этапов жизненного цикла ПО;
4. изучение методов верификации ПО;
5. приобретение навыков тестирования ПО в конкретной среде программирования.

2. Указания по проведению практических занятий

Тема: Жизненный цикл ПО

Практическое занятие 1.

Вид практического занятия: *смешанная форма практического занятия.*

Образовательная технология: *круглый стол*

Тема и содержание практического занятия:

Цель работы: Сформировать практические навыки реализации работ по созданию ПО.

Основные положения темы занятия:

1. Этапы жизненного цикла
2. Задачи этапа анализа

Вопросы для обсуждения:

1. Модели жизненного цикла.
2. Виды работ на каждом этапе жизненного цикла
3. CASE-средства анализа.

Продолжительность занятия – **2 ч.**

Тема: Методы анализа и проектирования ПО

Практическое занятие 2

Вид практического занятия: *смешанная форма практического занятия.*

Образовательная технология: *командная работа*

Тема и содержание практического занятия:

Цель работы: методы структурного проектирования

Основные положения темы занятия:

1. Принципы структурного проектирования
2. Нисходящее проектирование алгоритмов и программ
3. Разработка программы методом нисходящего проектирования

Продолжительность занятия - 6 ч.

Практическое занятие 3

Вид практического занятия: *смешанная форма практического занятия.*

Образовательная технология: *командная работа*

Тема и содержание практического занятия:

Цель работы: объектно-ориентированные методы анализа и проектирования.

Основные положения темы занятия:

1. Принципы объектно-ориентированного анализа и проектирования.
2. Анализ задачи и выделение классов
3. Продолжительность занятия - 2 ч.

Тема: Язык UML

Практическое занятие 4

Вид практического занятия: *смешанная форма практического занятия.*

Образовательная технология: *командная работа*

Тема и содержание практического занятия: Разработка UML диаграмм

Основные положения темы занятия:

1. Виды и назначение диаграмм UML.
2. Документирование диаграмм.
3. Инструментальные средства разработки диаграмм.

Задание на практическое занятие:

5. Разработать диаграмму вариантов использования и описать её.
6. Разработать диаграмму классов и описать её.
7. Разработать диаграмму взаимодействия и описать её.
8. Разработать диаграмму последовательностей и описать её.
9. Разработать диаграмму деятельности и описать её.
10. Разработать диаграмму состояний и описать её.
11. Продолжительность занятия - 8 ч.

Тема: Качество, надежность и эффективность ПО

Практическое занятие 5.

Вид практического занятия: *смешанная форма практического занятия.*

Образовательная технология: *командная работа*

Тема и содержание практического занятия:

Цель работы: с методами тестирования структурных алгоритмов и программ

Основные положения темы занятия:

1. Методы тестирования алгоритмов

2. Классификация ошибок в программе.
3. Средства отладки структурных программ
4. Стратегии черного и белого ящика.

Вопросы для обсуждения

1. Типы ошибок
2. Методы нахождения и исправления синтаксических ошибок
3. Статическое и динамическое тестирование.
4. Инспекция кода.
5. Тестовая база, тест, тестовый случай, тестовый монитор, тестовое покрытие и инструменты его измерения .
6. Проверка программы на нахождение ошибок с помощью метода «белого ящика».
7. Документирование программного обеспечения

Продолжительность занятия - 2 ч.

Практическое занятие 6.

Вид практического занятия: *смешанная форма практического занятия.*

Образовательная технология: *круглый стол*

Тема и содержание практического занятия:

Цель работы: Освоить методы тестирования объектно-ориентированных программ.

Основные положения темы занятия:

1. Качество программ
2. Характеристики и атрибуты качества.
3. Количественные оценки качества программного обеспечения

Вопросы для обсуждения

1. Средства тестирования
2. Ручное и автоматизированное тестирование
3. Применение методов и инструментальных средств тестирования.

Продолжительность занятия - 4 ч.

Тема: Интегрированные среды проектирования и разработки ПО

Практическое занятие 7.

Вид практического занятия: *смешанная форма практического занятия.*

Образовательная технология: *командная работа*

Тема и содержание практического занятия:

Цель работы: Получить практические знания по использованию языка C#.

Основные положения темы занятия:

1. Структура языка C#
2. Синтаксис операторов языка C#.

Задание на практическое занятие:

1. Разработать интерфейс программы в среде Microsoft Visual Studio на основе разработанной диаграммы вариантов использования
 2. Реализовать компоненты обработки данных
 3. Выполнить тестирование и отладку интерфейса и компонентов обработки данных.
 4. Разработать тестовые примеры для комплексного тестирования программы.
 5. Выполнить комплексное тестирование программы.
 6. Подготовить отчет по работе.
- Продолжительность занятия - 6 ч.

Практическое занятие 8.

Вид практического занятия: *смешанная форма практического занятия.*

Образовательная технология: *командная работа*

Тема и содержание практического занятия:

Цель работы: Получить практические знания по использованию технологии ADO.NET для разработки приложений с БД.

Основные положения темы занятия:

1. Методы доступа к удаленным данным в среде Microsoft Visual Studio.
2. Классы для работы с БД.
3. Технология LINQ

Задание на практическое занятие:

1. Разработать программные компоненты для реализации запросов к БД с использованием технологии LINQ.
 2. Разработать тестовые наборы.
 3. Выполнить тестирование и отладку компонентов.
 4. Выполнить комплексную отладку разработанных компонентов.
 5. Подготовить отчет по работе
- Продолжительность занятия - 2 ч.

3. Указания по проведению лабораторного практикума

Не предусмотрен учебным планом.

4. Указания по проведению самостоятельной работы студентов

Цель самостоятельной работы: подготовить бакалавров к самостоятельному научному творчеству.

Задачи самостоятельной работы:

1. Расширить представление о технологических методах разработки ПО.
2. Систематизировать знания в области проектирования ПО.
3. Сформировать навыки работы с программной документацией при разработке ПО.

Виды самостоятельной работы представлены в таблице 1.

Таблица 1

№ п/п	Наименование блока (раздела) дисциплины	Виды СРС
1	Тема 1. Жизненный цикл ПО	<p>Самостоятельное изучение тем с подготовкой презентаций</p> <p>Вопросы, выносимые на самостоятельное изучение:</p> <ol style="list-style-type: none"> 1. Задачи этапов жизненного цикла. 2. ISO/IEC 12207:1995 «Информационные технологии. Процессы жизненного цикла программного обеспечения» 3. ISO 15226:1999 «Техническая документация на продукцию. Модель жизненного цикла и назначение документов» 4. Стандарт ГОСТ 34.601. Автоматизированные системы. Стадии создания. 5. ISO/IEC 42010:2007 «Технология систем и программного обеспечения. Рекомендуемая практика архитектурного описания программно-интенсивных систем» 6. Характеристики качества ПО.
2	Тема 2. Методы анализа и проектирования ПО	<p>Самостоятельное изучение тем.</p> <p>Вопросы, выносимые на самостоятельное изучение:</p> <ol style="list-style-type: none"> 1. Классификация методов проектирования ПО. 2. Функциональные модели. <p>Примерная тематика докладов с презентацией:</p> <ol style="list-style-type: none"> 1. CASE-средства проектирования моделей ПО. 2. Инструментальные средства построения моделей ПО. 3. Технологии информационного моделирования.
3	Тема 3. Язык UML	<p>Самостоятельное изучение тем.</p> <p>Вопросы, выносимые на самостоятельное изучение:</p> <ol style="list-style-type: none"> 1. Инструментальные средства диаграмм UML 2. Элементы языка UML. <p>Примерная тематика докладов для презентаций:</p> <ol style="list-style-type: none"> 1. Применение диаграмм UML для решения практической задачи. 2. Документирование диаграмм UML
4	Тема 4. Качество, надежность и эффективность ПО	<p>Самостоятельное изучение тем.</p> <p>Вопросы, выносимые на самостоятельное изучение:</p> <ol style="list-style-type: none"> 1. Характеристика показателей качества ПО 2. Различие между тестированием и отладкой программ. 3. Средства тестирования <p>Примерная тематика рефератов:</p> <ol style="list-style-type: none"> 1. Методы отладки в различных программных

		<p>средах.</p> <p>2. Характеристика метода нисходящего тестирования.</p> <p>3. Пошаговое и монолитное тестирование модулей.</p>
5	<p>Тема 5. Интегрированные среды проектирования и разработки ПО</p>	<p>Самостоятельное изучение тем.</p> <p>Вопросы, выносимые на самостоятельное изучение:</p> <p>1. Основы работы в среде Microsoft Visual Studio.</p> <p>2. Типы проектов C#</p> <p>Примерная тематика докладов с презентацией:</p> <p>1. Технология создание проектов с использованием среды Microsoft Visual Studio.</p> <p>2. Назначение и описание класса</p> <p>3. Сравнительный анализ сред программирования</p>

5. Указания по проведению контрольных работ для студентов

Учебным планом для бакалавров предусмотрено написание одной контрольной работы, что является одним из условий успешного освоения ими основных положений данной дисциплины и служит допуском к сдаче экзамена по курсу во время зачетной сессии.

Задания в контрольной работе разрабатываются преподавателем кафедры «Информационных технологий и управляющих систем» ТУ.

Цель выполняемой работы: продемонстрировать знания и умения в области применения стандартов при разработке программных продуктов, а также показать умения в области систематизирования и обобщения изучаемой информации.

Основные задачи выполняемой работы:

1. Закрепление полученных теоретических знаний;
2. Получение навыков работы с технической документацией
3. Оценка применения практических навыков бакалавра в будущей практической работе;

Процесс написания контрольной работы делится на следующие этапы:

1. Определение темы контрольной работы
2. Изучение литературы, относящейся к теме контрольной работы
3. Оформление контрольной работы
4. Представление ее на кафедру для регистрации
5. Защита контрольной работы

Подготовку контрольной работы следует начинать с повторения соответствующих разделов учебника, учебных пособий, конспектов лекций.

Требования к содержанию контрольной работы:

В содержании контрольной работы необходимо показать знание рекомендованной литературы по данному заданию, при этом правильно пользоваться первоисточником и избегать чрезмерного цитирования. При использовании ци-

тат необходимо указывать точные ссылки на используемый источник: указание автора (авторов), название работы, место издания, страницы. Кроме основной литературы рекомендуется использовать дополнительную литературу и источники сети Интернет (с детальным указанием сайта, т.е. копирование ссылки и даты обращения). Если в период выполнения контрольной работы были приняты новые законы или нормативно-правовые акты, относящиеся к излагаемой теме, их необходимо изучить и использовать при ее выполнении. Важно обратить внимание на различные концептуальные подходы по исследуемой тематике.

Оформление библиографического списка осуществляется в соответствии с установленными нормами и правилами ГОСТ.

Порядок выполнения контрольной работы:

Контрольная работа излагается логически последовательно, грамотно, разборчиво.

Структура контрольной работы:

– титульный лист, который содержит полное название высшего учебного заведения, название кафедры, реализующей данную дисциплину, название (тема) контрольной работы, фамилию, инициалы автора, также необходимо указать номер группы, фамилию и инициалы, а также должность, ученое звание и степень научного руководителя (преподавателя), проверяющего контрольную работу.

- оглавление;
- введение;
- основная часть;
- заключение;
- список используемых источников;
- приложения.

В конце работы ставится подпись студента и дата сдачи. Страницы контрольной работы должны быть пронумерованы. Номер страницы ставится в нижнем правом углу.

Объем контрольной работы должен составлять 10-15 страниц машинописного текста. Размер шрифта №14 (Times New Roman), полуторный интервал, стандартный лист формата А4. Поля: верхнее -20 мм, нижнее-20мм, левое -30 мм, правое -15 мм.

Дополнительно контрольная работа может иметь приложения (схемы, графики, диаграммы).

По всем возникающим вопросам обучающемуся следует обращаться за консультацией на кафедру. Срок выполнения контрольной работы определяется кафедрой. Срок проверки контрольной работы – 3 дня с момента необходимой фиксированной даты сдачи.

Порядок защиты контрольной работы:

Контрольная работа подлежит обязательной защите. В установленной преподавателем срок студент должен сдать контрольную работу и быть готов ответить на вопросы и замечания. Оценка работы производится по четырех бальной системе: «ОТЛИЧНО», «ХОРОШО», «УДОВЛЕТВОРИТЕЛЬНО», «НЕУДО-

ВЛЕТВОРИТЕЛЬНО». После сдачи работы не возвращаются и хранятся в фонде кафедры.

Тематика контрольной работы:

1. Стандартизация разработки ПО
2. CASE-средства анализа и проектирования ПО.
3. Характеристики инструментальной среды программирования.
4. Технологии доступа к данным.
5. Технологии быстрой разработки ПО
6. Методы отладки программ в различных программных средах.
7. Характеристика метода нисходящего тестирования.
8. Пошаговое и монолитное тестирование модулей.
9. Сравнительный анализ интегрированных сред программирования.
10. Методология применения диаграмм UML для разработки программной системы.
11. CASE-средства проектирования моделей ПО.
12. Инструментальные средства построения моделей ПО.
13. Технологии информационного моделирования.
14. Сравнение моделей жизненного цикла программных систем.
15. Гибкие методологии разработки программных продуктов.
16. Задачи и проблемы программной инженерии.
17. Методологии разработки ПО.
18. Свойства сложных программных систем.
19. Структура программного обеспечения ИС
20. Проблемы оценки качества ПО
21. Средства информационной поддержки программных проектов и изделий (CALS) технологий
22. Классификация стандартов, определяющих разработку ПО.
23. Обзор стандарта ГОСТ Р ИСО/МЭК 12207-99.
24. Обзор стандарта ГОСТ Р ИСО 15288-2005
25. Обзор стандарта ГОСТ Р ИСО/МЭК ТО 15271-2002.
26. Стратегии разработки ПО.
27. Гибкие методологии разработки ПО.
28. Методология Microsoft разработки ПО
29. Методология IBM разработки ПО.
30. Методологии быстрой разработки ПО
31. Модели жизненного цикла.
32. Виды работ на каждом этапе жизненного цикла
33. CASE-средства анализа.
34. Принципы структурного проектирования
35. Нисходящее проектирование алгоритмов и программ
36. Разработка программы методом нисходящего проектирования
37. Объектно-ориентированный анализ.
38. Характеристика унифицированного процесса (Rational Unified Process, RUP) разработки ПО.
39. Характеристика Personal Software Process.
40. Уровни зрелости организаций.

41. Методологии проектирования и реализации распределенных приложений.
42. Преимущества объектно-ориентированного подхода к разработке ПО.
43. Сравнение стратегий разработки ПО
44. CASE-средства анализа и проектирования ПО.
45. Характеристики инструментальной среды программирования Microsoft Visual Studio.
46. Эволюция технологий доступа к данным.
47. Характеристика технологии .NET
48. Тенденции развития технологий программной инженерии.
49. Понятие проекта в Microsoft Visual Studio C#
50. Структура проекта в Microsoft Visual Studio c#
51. Шаблоны в в Microsoft Visual Studio C#

6. Перечень основной и дополнительной учебной литературы

Основная литература:

1. В.А. Антипов, А.А. Бубнов, А.Н. Пылькин, В.К. Столчнев. Введение в программную инженерию: Учебник. М.: КУРС: ИНФРА-М, 2022. — 336 с. ЭБС Знаниум. Режим доступа: <http://znanium.com/bookread2.php?book=850951>
2. Гагарина Л.Г., Кокорева Е.В., Сидорова-Виснадул Б.Д. Технология разработки программного обеспечения : учеб. пособие / под ред. Л.Г. Гагариной. — М.: ИД «ФОРУМ» : ИНФРА-М, 2023. — 400 с. / ЭБС «Знаниум» , режим доступа: <http://znanium.com/bookread2.php?book=924760>
3. Антамошкин О.А. Программная инженерия. Теория и практика. - 1. - Красноярск : Сибирский федеральный университет, 2021. - 247 с. - ISBN 9785763825114. - Электронная программа (визуальная). Электронные данные : электронные. URL: <http://znanium.com/go.php?id=492527>

Дополнительная литература:

1. Влацкая И.В., Заельская Е.А., Надточий Е.С. Проектирование и реализация прикладного программного обеспечения. – Оренбург: ОГУ. – 2023. – 119 с. ЭБС «Университетская библиотека онлайн». Режим доступа: http://biblioclub.ru/index.php?page=book_view_red&book_id=439107
2. Заботина Н.Н. Проектирование информационных систем : Учебное пособие. - 1. - Москва : ООО "Научно-издательский центр ИНФРА-М", 2021. - 331 с. - ISBN 9785160045092. - Электронная программа (визуальная). Электронные данные : электронные. URL: <http://znanium.com/go.php?id=1036508>

7. Перечень ресурсов информационно-телекоммуникационной сети «Интернет», необходимых для освоения дисциплины

Интернет-ресурсы:

1. <http://www.intuit.ru> - образовательный портал,
2. <http://www.infojournal.ru> - научно-образовательный портал,
3. <http://msdn.microsoft.com/ru-ru/library/> - портал разработчика Microsoft,
4. <http://www.interface.ru/> - научно-образовательный портал.

5. <http://www.biblioclub.ru> – электронная библиотечная система

6. <http://znanium.com> – электронная библиотечная система

8. Перечень информационных технологий

Перечень программного обеспечения: Microsoft Visio, Microsoft Visual Studio, Modelio Open Source.

Информационные справочные системы:

1. Ресурсы информационно-образовательной среды ТУ
2. Рабочая программа и методическое обеспечение по дисциплине «Введение в программную инженерию».