

Министерство науки и высшего образования Российской Федерации
Федеральное государственное бюджетное образовательное учреждение высшего образования «Ростовский государственный экономический университет (РИНХ)»
Документ подписан: 19.08.2024 11:45:25
Информация о владельце:
ФИО: Макаренко Елена Николаевна
Должность: Ректор
Уникальный программный ключ:
c098bc0c1041cb2a4cf926cf171d6715d99a6ae00adc8e27b55cbe1e2dbd7c78

УТВЕРЖДАЮ
Директор Института магистратуры
Иванова Е.А.
«03» июня 2024г.

**Рабочая программа дисциплины
Программная инженерия: управление разработкой**

Направление 09.04.04 Программная инженерия
магистерская программа 09.04.04.01 "Системное и прикладное программное обеспечение"

Для набора 2024 года

Квалификация
магистр

КАФЕДРА **Информационные технологии и программирование****Распределение часов дисциплины по семестрам**

Семестр (<Курс>.<Семестр на курсе>)	2 (1.2)		3 (2.1)		Итого	
	Неделя		15 2/6			
Вид занятий	УП	РП	УП	РП	УП	РП
Лабораторные	16	16	16	16	32	32
Практические	8	8	16	16	24	24
Итого ауд.	24	24	32	32	56	56
Контактная работа	24	24	32	32	56	56
Сам. работа	120	120	112	112	232	232
Часы на контроль	36	36	36	36	72	72
Итого	180	180	180	180	360	360

ОСНОВАНИЕ

Учебный план утвержден учёным советом вуза от 26.03.2024 протокол № 13.

Программу составил(и): к.ф.-м.н, доц., Карнаухов С.Н.

Зав. кафедрой: к.э.н., доц. Ефимова Е.В.

Методическим советом направления: д.э.н., проф., Тищенко Е.Н.

1. ЦЕЛИ ОСВОЕНИЯ ДИСЦИПЛИНЫ

- 1.1 Формирование практических навыков управления разработкой программного обеспечения на всех этапах ЖЦ с применением современных информационных технологий

2. ТРЕБОВАНИЯ К РЕЗУЛЬТАТАМ ОСВОЕНИЯ ДИСЦИПЛИНЫ

УК-4:Способен применять современные коммуникативные технологии, в том числе на иностранном(ых) языке(ах), для академического и профессионального взаимодействия

ОПК-6:Способен самостоятельно приобретать с помощью информационных технологий и использовать в практической деятельности новые знания и умения, в том числе в новых областях знаний, непосредственно не связанных со сферой деятельности;

ОПК-7:Способен применять при решении профессиональных задач методы и средства получения, хранения, переработки и трансляции информации посредством современных компьютерных технологий, в том числе, в глобальных компьютерных сетях;

В результате освоения дисциплины обучающийся должен:

Знать:

современные коммуникативные технологии на государственном и иностранном языках; закономерности деловой устной и письменной коммуникации (соотнесено с индикатором УК-4.1).
информационные технологии для использования в практической деятельности (соотнесено с индикатором ОПК-6.1)
методы и средства получения, хранения, переработки и трансляции информации посредством современных компьютерных технологий, в том числе, в глобальных компьютерных сетях (соотнесено с индикатором ОПК-7.1)

Уметь:

применять на практике коммуникативные технологии, методы и способы делового общения (соотнесено с индикатором УК- 4.2).
самостоятельно приобретать новые знания и умения (соотнесено с индикатором ОПК-6.2)
применять методы и средства получения, хранения, переработки и трансляции информации посредством современных компьютерных технологий, в том числе, в глобальных компьютерных сетях (соотнесено с индикатором ОПК-7.2)

Владеть:

методикой межличностного делового общения на государственном и иностранном языках, с применением профессиональных языковых форм и средств (соотнесено с индикатором УК-4.3).
самостоятельно приобретать новые знания и умения в новых областях знаний (соотнесено с индикатором ОПК-6.3)
методы и средства получения, хранения, переработки и трансляции информации посредством современных компьютерных технологий, в том числе, в глобальных компьютерных сетях (соотнесено с индикатором ОПК-7.3)

3. СТРУКТУРА И СОДЕРЖАНИЕ ДИСЦИПЛИНЫ

Раздел 1. Управление разработкой ПО: основные понятия

№	Наименование темы / Вид занятия	Семе стр	Часов	Компетен- ции	Литература
1.1	Тема 1.1 Унифицированный язык моделирование UML. Диаграммы UML. Основные принципы визуального моделирования. Сложность ПО и архитектурные представления. Статические и динамические диаграммы UML. Репозиторий модели CASE средства. Прямое и обратное проектирование кода программ и структуры базы данных. Стереотипы UML и их использование / Лаб /	2	4	ОПК-7,УК-4,ОПК-6	Л1.1, Л1.2, Л2.1, Л2.2, Л2.3
1.2	Тема 1.1.Планирование коллективной разработки ПО. Системный подход. Каскадная и спиральная модель жизненного цикла ПО. Инкрементная модель ЖЦ разработки. Методика выбора модели ЖЦ разработки ПП. Технология управления изменениями. Разработка концепций программного проекта, выбор перспективной концепции. Декомпозиция (разбиение) разработки ПО на подсистемы – универсальный метод снижения сложности разработки. Аутсорсинг / Пр /	2	4	ОПК-7,УК-4,ОПК-6	Л1.1, Л1.2, Л2.1, Л2.2, Л2.3
1.3	Тема 1.1. Управление требованиями. Виды требований: функциональные требования,	2	4	ОПК-7,УК-4,ОПК-6	Л1.1, Л1.2, Л2.1, Л2.2, Л2.3

	нефункциональные требования. Свойства требований: ясность и недвусмысленность, полнота и непротиворечивость, необходимый уровень детализации, прослеживаемость, тестируемость и проверяемость, модифицируемость. Формализация требований. Цикл работы с требованиями. Типы UML диаграмм. / Лаб /				
1.4	Введение в технологии разработки программного обеспечения. Понятие процесса разработки ПО. Универсальный процесс. Текущий процесс. Конкретный процесс. Стандартный процесс. Совершенствование процесса. Pull/Push стратегии. Классические модели процесса: водопадная модель, спиральная модель. Фазы и виды деятельности Процессы командной разработки программного обеспечения. Гибкие технологии разработки ПО. Операционная и проектная деятельность. Метод функционального описания предметной области. Понятие модели. Нотация структурного функционального моделирования IDEF0. Средства разработки процессных моделей. / Ср /	2	60	ОПК-7,УК-4,ОПК-6	Л1.1, Л1.2, Л2.1, Л2.2, Л2.3

Раздел 2. Технология коллективной разработки программного обеспечения

№	Наименование темы / Вид занятия	Семе стр	Часов	Компетенции	Литература
2.1	Тема 2.1 Освоение компонентов интерфейса и настройка среды Draw.io предназначенной для обеспечения совместной работы команд разработчиков ПО. Подключение к TFS. Создание командного проекта. / Лаб /	2	4	ОПК-7,УК-4,ОПК-6	Л1.1, Л1.2, Л2.1, Л2.2, Л2.3
2.2	Тема 2.1 Управление жизненным циклом приложений. Архитектурное проектирование. Архитектура и функциональные возможности Draw.io / Пр /	2	4	ОПК-7,УК-4,ОПК-6	Л1.1, Л1.2, Л2.1, Л2.2, Л2.3
2.3	Тема 2.2 Шаблоны командных проектов. MSFforCMMIProcessImprovement 6.0 MSFforAgileSoftwareDevelopment 6.0 Draw.io Анализ системных требований для установки Draw.io / Лаб /	2	4	ОПК-7,УК-4,ОПК-6	Л1.1, Л1.2, Л2.1, Л2.2, Л2.3
2.4	Архитектурное проектирование. Архитектура и функциональные возможности Visual Studio Code / Ср /	2	60	ОПК-7,УК-4,ОПК-6	Л1.1, Л1.2, Л2.1, Л2.2, Л2.3
2.5	/ Экзамен /	2	36	ОПК-7,УК-4,ОПК-6	Л1.1, Л1.2, Л2.1, Л2.2, Л2.3

Раздел 3. Управление рисками процесса разработки

№	Наименование темы / Вид занятия	Семе стр	Часов	Компетенции	Литература
3.1	Тема 3.1 Управление рисками процесса разработки ПО в программной среде Visual Studio Code / Лаб /	3	4	ОПК-7,УК-4,ОПК-6	Л1.1, Л1.2, Л2.1, Л2.2, Л2.3
3.2	Тема 3.1 Практическая организация и управление рисками процесса разработки ПО в программной среде Visual Studio Code / Пр /	3	6	ОПК-7,УК-4,ОПК-6	Л1.1, Л1.2, Л2.1, Л2.2, Л2.3
3.3	Тема 3.2 Управление рисками с помощью процесса СММИ / Лаб /	3	4	ОПК-7,УК-4,ОПК-6	Л1.1, Л1.2, Л2.1, Л2.2, Л2.3
3.4	Основные понятия риска и риск образующих факторов. Управление рисками на каждом из этапов ЖЦ программного проекта: идентификацию рисков, анализ рисков, планирование управления рисками, мониторинг и методы реагирования на риски. Наиболее распространенные риски программных проектов. Методы контроля хода исполнения программных проектов / Ср /	3	52	ОПК-7,УК-4,ОПК-6	Л1.1, Л1.2, Л2.1, Л2.2, Л2.3

Раздел 4. Создание и управление проектами с использованием Team Foundation Server

№	Наименование темы / Вид занятия	Семестр	Часов	Компетенции	Литература
4.1	Тема 4.1 Создание проекта Выбор шаблона процесса. Создание группового проекта. . Определение того, какие сценарии должны быть завершены в каждой итерации. Определение нефункциональных требований. Связь нефункциональных требований со сценариями. Разбиение сценариев на истории Определение критериев приемки. Определение требований к отчетам. / Лаб /	3	4	ОПК-7,УК-4,ОПК-6	Л1.1, Л1.2, Л2.1, Л2.2, Л2.3
4.2	Тема 4.1 Создание графика выполнения проекта. Создается график выполнения проекта и добавляется в Team Project. / Пр /	3	2	ОПК-7,УК-4,ОПК-6	Л1.1, Л1.2, Л2.1, Л2.2, Л2.3
4.3	Тема 4.2 Добавление нового разработчика в проект / Лаб /	3	2	ОПК-7,УК-4,ОПК-6	Л1.1, Л1.2, Л2.1, Л2.2, Л2.3
4.4	Тема 4.2 Анализ кода при помощи Team Build тестирование сборки включение анализа кода в сборку тестирование анализа кода / Пр /	3	4	ОПК-7,УК-4,ОПК-6	Л1.1, Л1.2, Л2.1, Л2.2, Л2.3
4.5	Тема 4.3 Создание отчета в Visual Studio Code создание нового проекта отчетов. создание источников данных. создание нового отчета в проекте. изменение отчета. развертывание отчета на Team Foundation Server. тестирование отчета. / Пр /	3	4	ОПК-7,УК-4,ОПК-6	Л1.1, Л1.2, Л2.1, Л2.2, Л2.3
4.6	Тема 4.3 создание пользовательской политики возврата после правки создание и сборка класса пользовательской политики. регистрация класса пользовательской политики в реестре. применение пользовательской политики. проверка работоспособности пользовательской политики. / Лаб /	3	2	ОПК-7,УК-4,ОПК-6	Л1.1, Л1.2, Л2.1, Л2.2, Л2.3
4.7	Обзор методологий разработки ПО. MSF. IT решение. Основные принципы MSF. Модель команды: основные принципы, ролевые кластеры. Масштабирование команды MSF. Модель процесса. Уровни зрелости процессов по СММІ. Области усовершенствования. "Гибкие" (agile) методы разработки. Общее описание "гибких" методов разработки ПО. Extreme Programming: общее описание, основные принципы организации процесса. Scrum: общее описание, роли, практики / Ср /	3	60	ОПК-7,УК-4,ОПК-6	Л1.1, Л1.2, Л2.1, Л2.2, Л2.3
4.8	/ Экзамен /	3	36	ОПК-7,УК-4,ОПК-6	Л1.1, Л1.2, Л2.1, Л2.2, Л2.3

4. ФОНД ОЦЕНОЧНЫХ СРЕДСТВ

Структура и содержание фонда оценочных средств для проведения текущей и промежуточной аттестации представлены в Приложении 1 к рабочей программе дисциплины.

5. УЧЕБНО-МЕТОДИЧЕСКОЕ И ИНФОРМАЦИОННОЕ ОБЕСПЕЧЕНИЕ ДИСЦИПЛИНЫ

5.1. Основная литература

	Авторы, составители	Заглавие	Издательство, год	Колич-во
Л1.1		Информационная безопасность: журнал	Москва: Гротек, 2014	https://biblioclub.ru/index.php?page=book&id=364894 неограниченный доступ для зарегистрированных пользователей

	Авторы, составители	Заглавие	Издательство, год	Колич-во
Л1.2	Носова, Л. С.	Case-технологии и язык UML: учебно-методическое пособие	Челябинск, Саратов: Южно-Уральский институт управления и экономики, Ай Пи Эр Медиа, 2019	https://www.iprbookshop.ru/81479.html неограниченный доступ для зарегистрированных пользователей

5.2. Дополнительная литература

	Авторы, составители	Заглавие	Издательство, год	Колич-во
Л2.1	Лисяк В. В.	Разработка информационных систем: учебное пособие	Ростов-на-Дону, Таганрог: Южный федеральный университет, 2019	https://biblioclub.ru/index.php?page=book&id=577875 неограниченный доступ для зарегистрированных пользователей
Л2.2	Орлов С. А., Цилькер Б. Я.	Технологии разработки программного обеспечения. Учебник для вузов. 4-е издание. Стандарт третьего поколения	Санкт-Петербург: Питер, 2012	https://ibooks.ru/reading.php?short=1&productid=28460 неограниченный доступ для зарегистрированных пользователей
Л2.3	Орлов С. А.	Программная инженерия. Учебник для вузов. 5-е издание обновленное и дополненное. Стандарт третьего поколения. — (Серия «Учебник для вузов»).	Санкт-Петербург: Питер, 2021	https://ibooks.ru/reading.php?short=1&productid=376973 неограниченный доступ для зарегистрированных пользователей

5.3 Профессиональные базы данных и информационные справочные системы

Официальная документация Microsoft Visual Studio Team Foundation Server(TFS) <https://www.microsoft.com/ru-ru/download/developer-tools.aspx>

Официальная документация Draw.io <https://github.com/jgraph/drawio-desktop/releases/tag/v20.8.16>

ИСС «КонсультантПлюс»

ИСС «Гарант» <http://www.internet.garant.ru/>

5.4. Перечень программного обеспечения

Draw.io – это допустимое ПО?

Visual Studio Code

5.5. Учебно-методические материалы для студентов с ограниченными возможностями здоровья

При необходимости по заявлению обучающегося с ограниченными возможностями здоровья учебно-методические материалы предоставляются в формах, адаптированных к ограничениям здоровья и восприятия информации. Для лиц с нарушениями зрения: в форме аудиофайла; в печатной форме увеличенным шрифтом. Для лиц с нарушениями слуха: в форме электронного документа; в печатной форме. Для лиц с нарушениями опорно-двигательного аппарата: в форме электронного документа; в печатной форме.

6. МАТЕРИАЛЬНО-ТЕХНИЧЕСКОЕ ОБЕСПЕЧЕНИЕ ДИСЦИПЛИНЫ (МОДУЛЯ)

Помещения для всех видов работ, предусмотренных учебным планом, укомплектованы необходимой специализированной учебной мебелью и техническими средствами обучения:

- столы, стулья;

- персональный компьютер / ноутбук (переносной);

- проектор;

- экран / интерактивная доска

Лабораторные занятия проводятся в компьютерных классах, рабочие места в которых оборудованы необходимыми лицензионными и/или свободно распространяемыми программными средствами и выходом в Интернет.

7. МЕТОДИЧЕСКИЕ УКАЗАНИЯ ДЛЯ ОБУЧАЮЩИХСЯ ПО ОСВОЕНИЮ ДИСЦИПЛИНЫ (МОДУЛЯ)

Методические указания по освоению дисциплины представлены в Приложении 2 к рабочей программе дисциплины.

ФОНД ОЦЕНОЧНЫХ СРЕДСТВ

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

1.1 Показатели и критерии оценивания компетенций:

ЗУН, составляющие компетенцию	Показатели оценивания	Критерии оценивания	Средства оценивания
УК-4: Способен применять современные коммуникативные технологии, в том числе на иностранном(ых) языке(ах), для академического и профессионального взаимодействия			
3. современные коммуникативные технологии на государственном и иностранном языках; закономерности деловой устной и письменной коммуникации (соотнесено с индикатором УК-4.1).	выбор методов распределенного хранения и обработки данных для решения профессиональных задач	полнота и обоснованность выбора методов распределенного хранения и обработки данных на основе изученной литературы	УО (Раздел 1 вопросы 1-37, Раздел 2 вопросы 1-10, Раздел 3 вопросы 1-7, Раздел 4 вопросы 1-27) Э (2 семестр вопросы 1-46, 3 семестр вопросы 1-28)
У. применять на практике коммуникативные технологии, методы и способы делового общения. (соотнесено с индикатором УК-4.2).	решение практико-ориентированных и лабораторных заданий на основе официальной документации: UML	правильность применения технологий, описанных в официальной документации UML	ЛЗ (1-8) ПЗ (1-5) ПОЭ (1, 2, 3,4)
В. методикой межличностного делового общения на государственном и иностранном языках, с применением профессиональных языковых форм и средств. (соотнесено с индикатором УК-4.3).	использование различных подходов к решению задач при построение UML диаграмм; применяет технологии коллективной разработки программного обеспечения	правильность и обоснованность различных подходов к решению задач при построение UML диаграмм; применяет технологии коллективной разработки программного обеспечения для решения профессиональных задач	ЛЗ (1-8) ПЗ (1-5) ПОЭ (1, 2, 3,4)
ОПК-6:Способен самостоятельно приобретать с помощью информационных технологий и использовать в практической деятельности новые знания и умения, в том числе в новых областях знаний, непосредственно не связанных со сферой деятельности;			
3. информационные технологии для использования в практической деятельности (соотнесено с индикатором ОПК-6.1)	выбор информационных технологий для решения профессиональных задач	полнота и обоснованность выбора информационных технологий на основе изученной литературы	УО (Раздел 1 вопросы 1-37, Раздел 2 вопросы 1-10, Раздел 3 вопросы 1-7, Раздел 4 вопросы 1-27) Э (2 семестр вопросы 1-46, 3 семестр вопросы 1-28)

У. - самостоятельно приобретать новые знания и умения (соотнесено с индикатором ОПК-6.2)	решение практико-ориентированных и лабораторных заданий: применяет высокоуровневые языки и инструментарию; осуществляет практическую организацию и управление рисками процесса разработки ПО в программной среде	правильность применения инструментария для практической организации и управления рисками процесса разработки ПО в программной среде при решении различных технических задач и проблем.	ЛЗ (1-8) ПЗ (1-5) ПОЭ (1, 2, 3,4)
В. самостоятельно приобретать новые знания и умения в новых областях знаний (соотнесено с индикатором ОПК-6.3)	решение практико-ориентированных и лабораторных заданий различными способами высокоуровневые языки и инструментарию; осуществляет практическую организацию и управление рисками процесса разработки ПО в программной среде для решения различных технических задач и проблем.	правильность и обоснованность выбора инструментария для практической организации и управления рисками процесса разработки ПО в программной при решении различных технических задач и проблем.	ЛЗ (1-8) ПЗ (1-5) ПОЭ (1, 2, 3,4)
ОПК-7:Способен применять при решении профессиональных задач методы и средства получения, хранения, переработки и трансляции информации посредством современных компьютерных технологий, в том числе, в глобальных компьютерных сетях;			
3. методы и средства получения, хранения, переработки и трансляции информации посредством современных компьютерных технологий, в том числе, в глобальных компьютерных сетях (соотнесено с индикатором ОПК-7.1)	выбор методов получения, хранения, переработки и трансляции информации посредством современных компьютерных технологий, в том числе, в глобальных компьютерных сетях для решения профессиональных задач	полнота и обоснованность выбора методов получения, хранения, переработки и трансляции информации посредством современных компьютерных технологий, в том числе, в глобальных компьютерных сетях основе изученной литературы	УО (Раздел 1 вопросы 1-37, Раздел 2 вопросы 1-10, Раздел 3 вопросы 1-7, Раздел 4 вопросы 1-27) Э (2 семестр вопросы 1-46, 3 семестр вопросы 1-28)
У. применять методы и средства получения, хранения, переработки и трансляции информации посредством современных компьютерных технологий, в том числе, в глобальных компьютерных сетях (соотнесено с индикатором ОПК-7.2)	Создание и управление проектами с использованием Team Foundation Server, (Выбор шаблона процесса,. создание группового проекта, определение требований, добавление нового разработчика в проект; создание отчета)	правильность применения инструментария: Создание и управление проектами с использованием Team Foundation Server	ЛЗ (1-8) ПЗ (1-5) ПОЭ (1, 2, 3,4)
В. методы и средства получения, хранения, переработки и трансляции информации посредством современных	Различными способами осуществляет Создание и управление проектами с использованием Team Foundation Server, (Выбор шаблона процесса,.	правильность и обоснованность выбора инструментария при создании и управлении проектами с использованием Team	ЛЗ (1-8) ПЗ (1-5) ПОЭ (1, 2, 3,4)

компьютерных технологий, в том числе, в глобальных компьютерных сетях (соотнесено с индикатором ОПК-7.3)	создание группового проекта, определение требований, добавление нового разработчика в проект; создание отчета)	Foundation Server, (Выбор шаблона процесса,. создание группового проекта, определение требований, добавление нового разработчика в проект; создание отчета)	
--	--	---	--

*З – вопросы к экзамену, ЛЗ – лабораторное задание, ПЗ - Практическое задание
 ПОЭ-практико-ориентированное задание к экзамену, УО- устный опрос*

1.2 Шкалы оценивания:

Текущий контроль успеваемости и промежуточная аттестация осуществляется в рамках накопительной балльно-рейтинговой системы в 100-балльной шкале:

50-100 баллов (зачет)

0-49 баллов (незачет)

2 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

В разделе приводятся типовые варианты оценочных средств: вопросы к зачету, практико-ориентированные задания к зачету, вопросы для устного опроса, лабораторные задания, практические задания,

Вопросы к экзамену 2 семестр

1. Унифицированный язык моделирование UML. Диаграммы UML.
2. Основные принципы визуального моделирования.
3. Сложность ПО и архитектурные представления.
4. Статические и динамические диаграммы UML.
5. Репозиторий модели CASE средства.
6. Прямое и обратное проектирование кода программ и структуры базы данных.
7. Стереотипы UML и их использование
8. Планирование коллективной разработки ПО.
9. Системный подход.
10. Каскадная и спиральная модель жизненного цикла ПО.
11. Инкрементная модель ЖЦ разработки.
12. Методика выбора модели ЖЦ разработки ПП.
13. Технология управления изменениями.
14. Разработка концепций программного проекта, выбор перспективной концепции.
15. Декомпозиция (разбиение) разработки ПО на подсистемы – универсальный метод снижения сложности разработки.
16. Аутсорсинг
17. Управление требованиями.
18. Виды требований: функциональные требования, нефункциональные требования.
19. Свойства требований
20. Формализация требований. Цикл работы с требованиями.
21. Понятие процесса разработки ПО.
22. Универсальный процесс.
23. Текущий процесс.
24. Конкретный процесс.
25. Стандартный процесс.
26. Совершенствование
27. процесса.
28. Pull/Push стратегии.

29. Классические модели процесса: водопадная модель, спиральная модель.
30. Фазы и виды деятельности
31. Процессы командной разработки программного обеспечения.
32. Гибкие технологии разработки ПО.
33. Операционная и проектная деятельность.
34. Метод функционального описания предметной области.
35. Нотация структурного функционального моделирования IDEF0.
36. Средства разработки процессных моделей.
37. Освоение компонентов интерфейса и настройка среды предназначенной для обеспечения совместной работы команд разработчиков ПО.
38. Подключение к TFS.
39. Создание командного проекта.
40. Управление жизненным циклом приложений.
41. Архитектурное проектирование.
42. Архитектура и функциональные возможности
43. Шаблоны командных проектов.
44. Анализ системных требований для установки
45. Анализ процесса установки и конфигурирования TeamFoundationServer2012

3 семестр

1. Управление рисками процесса разработки ПО в программной среде
2. Практическая организация и управление рисками процесса разработки ПО в программной среде
3. Управление рисками с помощью процесса СММІ
4. Основные понятия риска и риск образующих факторов.
5. Управление рисками на каждом из этапов ЖЦ программного проекта: идентификация рисков, анализ рисков, планирование управления рисками, мониторинг и методы реагирования на риски.
6. Наиболее распространенные риски программных проектов.
7. Методы контроля хода исполнения программных проектов
8. Создание проекта
9. Выбор шаблона процесса.
10. Создание группового проекта. .
11. Определение того, какие сценарии должны быть завершены в каждой итерации.
12. Определение нефункциональных требований. Связь нефункциональных требований со сценариями.
13. Разбиение сценариев на истории
14. Определение критериев приемки.
15. Определение требований к отчетам.
16. Создание графика выполнения проекта.
17. Добавление нового разработчика в проект
18. Анализ кода при помощи Team Build
19. Создание отчета в Visual Studio 2005 Team Foundation Server
20. создание пользовательской политики возврата после правки
21. Обзор методологий разработки ПО. MSF. IT решение.
22. Основные принципы MSF.
23. Модель команды: основные принципы, ролевые кластеры.
24. Масштабирование команды MSF.
25. Модель процесса.
26. Уровни зрелости процессов по СММІ.
27. Области усовершенствования. "Гибкие" (agile) методы разработки. Общее описание "гибких" методов разработки ПО.
28. Extreme Programming: общее описание, основные принципы организации процесса. Scrum: общее описание, роли, практики

Практико-ориентированные задания

Задание 1

Создание графика выполнения проекта. Создается график выполнения проекта и добавляется в Team Project

Задание 2

Выбор, обоснование и применение шаблона командных проектов.

MSFforCMMIProcessImprovement 6.0

MSFforAgileSoftwareDevelopment 6.0

Задание 3

Анализ кода при помощи Team Build (тестирование сборки, включение анализа кода в сборку, тестирование анализа кода)

Задание 4

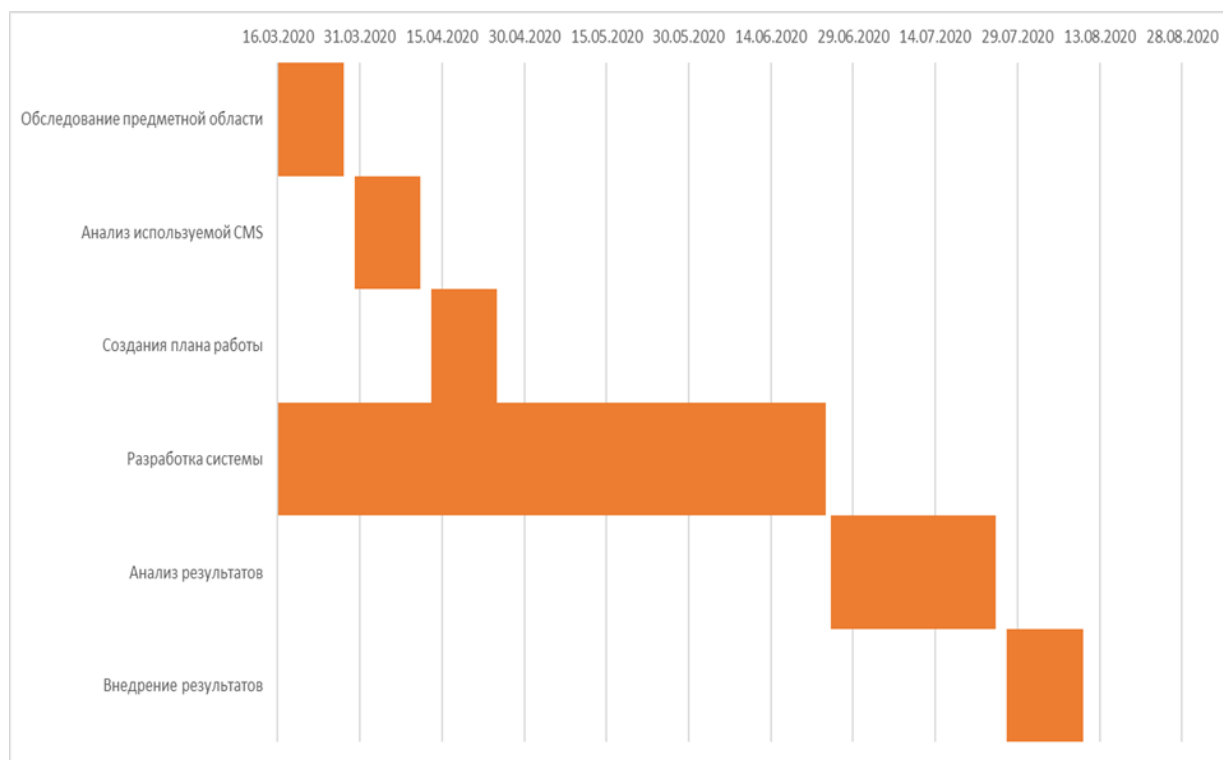
Создание нового отчета в проекте, изменение отчета.

Развертывание отчета на Team Foundation Server.

Тестирование отчета

Ключ для контроля правильности практико-ориентированного задания к экзамену

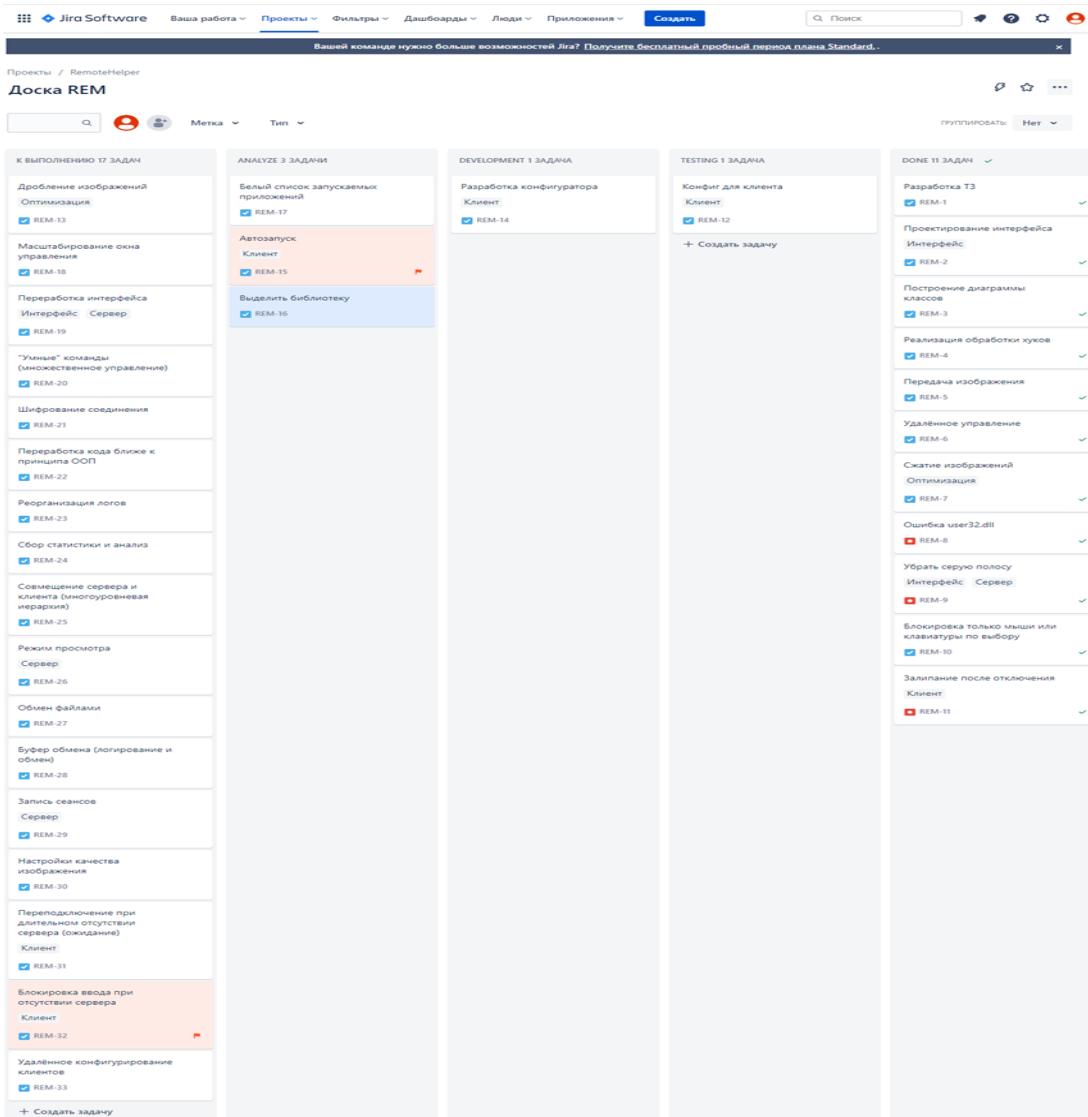
1. Создание графика выполнения проекта. Создается график выполнения проекта



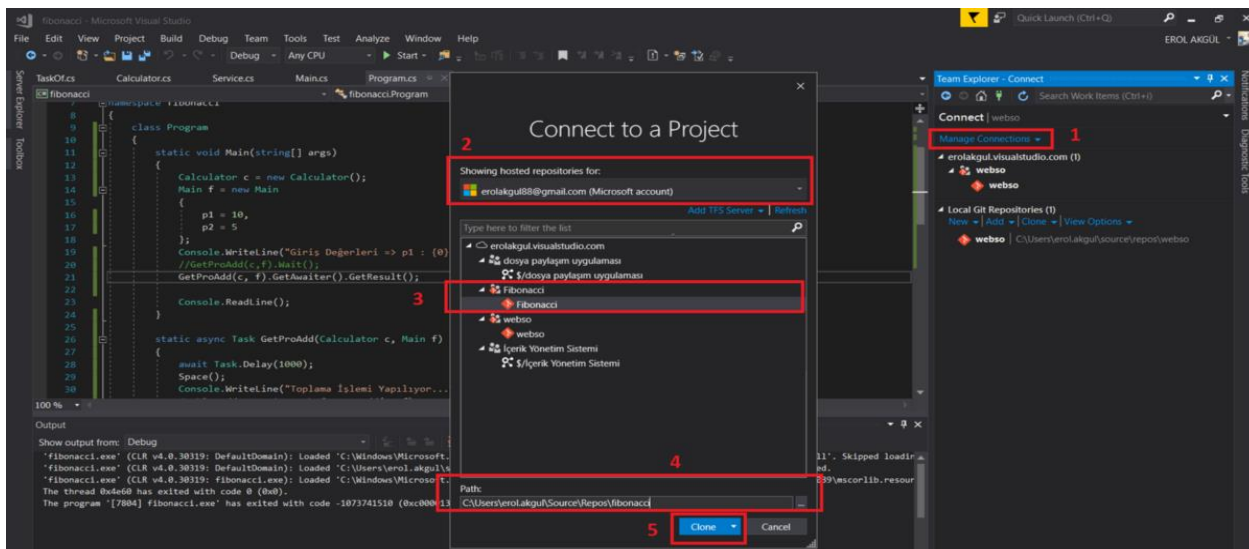
2. Выбор, обоснование и применение шаблона командных проектов.

MSFforCMMIProcessImprovement 6.0

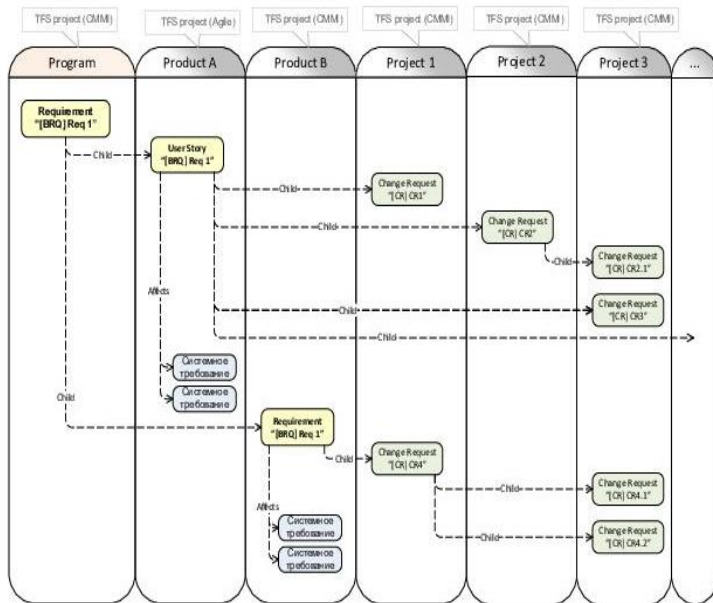
MSFforAgileSoftwareDevelopment 6.0



3. Анализ кода при помощи Team Build



4. Создание нового отчета в проекте, изменение отчета. Развертывание отчета на Team Foundation Server



27

Критерии оценивания:

- 50-100 баллов (зачет) – изложенный материал фактически верен, наличие глубоких исчерпывающих знаний; правильные, уверенные действия по применению полученных знаний на практике, грамотное и логически стройное изложение материала при ответе; практико-ориентированное задание выполнено правильно и прокомментировано; наличие твердых и достаточно полных знаний, правильные действия по применению знаний на практике, четкое изложение материала, допускаются отдельные логические и стилистические погрешности, неуверенность и неточность ответов на дополнительные и наводящие вопросы; практико-ориентированное задание выполнено правильно, но не прокомментировано; при неполном ответе на вопросы; затрудняется ответить на дополнительные вопросы; практико-ориентированное задание выполнено с ошибками и отсутствуют комментарии;
- 0-49 баллов (незачет) – ответы не связаны с вопросами, наличие грубых ошибок в ответе, непонимание сущности излагаемого вопроса, неумение применять знания на практике, неуверенность и неточность ответов на дополнительные и наводящие вопросы; практико-ориентированное задание не выполнено.

Лабораторные задания 2 семестр

Раздел 1. Управление разработкой ПО: основные понятия

Лабораторное задание 1 (10 баллов).

Тема 1.1 Унифицированный язык моделирование UML. Диаграммы UML.

Основные принципы визуального моделирования. Сложность ПО и архитектурные представления.

Статические и динамические диаграммы UML. Репозиторий модели CASE средства. Прямое и обратное проектирование кода программ и структуры базы данных. Стереотипы UML и их использование

Лабораторное задание 2 (20 баллов).

Тема 1.2. Управление требованиями. Виды требований: функциональные требования, нефункциональные требования. Свойства требований: ясность и недвусмысленность, полнота и непротиворечивость, необходимый уровень детализации, прослеживаемость, тестируемость и проверяемость, модифицируемость. Формализация требований. Цикл работы с требованиями. Типы UML диаграмм.

Раздел 2. Технология коллективной разработки программного обеспечения

Лабораторное задание 3 (10 баллов).

Тема 2.1 Освоение компонентов интерфейса и настройка среды Microsoft Visual Studio Team Foundation Server (TFS) предназначенной для обеспечения совместной работы команд разработчиков ПО. Подключение к TFS. Создание командного проекта.

Лабораторное задание 4 (20 баллов).

Тема 2.2 Шаблоны командных проектов.

MSFforCMMIProcessImprovement 6.0

MSFforAgileSoftwareDevelopment 6.0

Анализ системных требований для установки Visual Studio Code TeamFoundationServer

Анализ процесса установки и конфигурирования TeamFoundationServer2012.

Максимальное количество баллов по лабораторным заданиям – 60

3 семестр

Раздел 3. Управление рисками процесса разработки

Лабораторное задание 5 (8 баллов).

Тема 3.1 Управление рисками процесса разработки ПО в программной среде Visual Studio Code Team Foundation Server

Лабораторное задание 6 (12 баллов).

Тема 3.2 Управление рисками с помощью процесса СММІ

Раздел 4. Создание и управление проектами с использованием Team Foundation Server

Лабораторное задание 7 (8 баллов).

Создание проекта

Выбор шаблона процесса.

Создание группового проекта. .

Определение того, какие сценарии должны быть завершены в каждой итерации.

Определение нефункциональных требований. Связь нефункциональных требований со сценариями.

Разбиение сценариев на истории

Определение критериев приемки.

Определение требований к отчетам

Лабораторное задание 8 (12 баллов).

Тема 4.2 Добавление нового разработчика в проект Visual Studio Code Team Foundation Server

Максимальное количество баллов по лабораторным заданиям – 40

Практические задания

2 семестр

Раздел 1. Управление разработкой ПО: основные понятия

Практическое задание 1 (10 баллов).

Тема 1.1. Планирование коллективной разработки ПО. Системный подход. Каскадная и спиральная модель жизненного цикла ПО. Инкрементная модель ЖЦ разработки. Методика выбора модели ЖЦ разработки ПП. Технология управления изменениями. Разработка концепций программного проекта, выбор перспективной концепции. Декомпозиция (разбиение) разработки ПО на подсистемы – универсальный метод снижения сложности разработки. Аутсорсинг

Раздел 2. Технология коллективной разработки программного обеспечения

Практическое задание 1 (10 баллов).

Тема 2.1 Управление жизненным циклом приложений. Архитектурное проектирование. Архитектура и функциональные возможности Visual Studio Code TeamFoundation Server (TFS)

Максимальное количество баллов по практическим заданиям – 20

3 семестр

Раздел 3. Управление рисками процесса разработки

Практическое задание 2 (8 баллов).

Тема 3.1 Практическая организация и управление рисками процесса разработки ПО в программной среде **Visual Studio Code**

Раздел 4. Создание и управление проектами с использованием Team Foundation Server

Практическое задание 3 (5 баллов).

Тема 4.1 Создание графика выполнения проекта. Создается график выполнения проекта и добавляется в Team Project

Практическое задание 4 (7 баллов).

Анализ кода при помощи Team Build

тестирование сборки

включение анализа кода в сборку

тестирование анализа кода

Практическое задание 5 (8 баллов).

Создание отчета в Visual Studio Code

создание нового проекта отчетов.

создание источников данных.

создание нового отчета в проекте.

изменение отчета.

развертывание отчета на Team Foundation Server.

тестирование отчета

Максимальное количество баллов по практическим заданиям – 20

Перечень вопросов для устного опроса

Раздел 1. Управление разработкой ПО: основные понятия

1. Унифицированный язык моделирование UML. Диаграммы UML.
2. Основные принципы визуального моделирования.
3. Сложность ПО и архитектурные представления.
4. Статические и динамические диаграммы UML.
5. Репозиторий модели CASE средства.
6. Прямое и обратное проектирование кода программ и структуры базы данных.
7. Стереотипы UML и их использование
8. Планирование коллективной разработки ПО.
9. Системный подход.
10. Каскадная и спиральная модель жизненного цикла ПО.
11. Инкрементная модель ЖЦ разработки.
12. Методика выбора модели ЖЦ разработки ПП.
13. Технология управления изменениями.
14. Разработка концепций программного проекта, выбор перспективной концепции.
15. Декомпозиция (разбиение) разработки ПО на подсистемы – универсальный метод снижения сложности разработки.
16. Аутсорсинг
17. Управление требованиями.
18. Виды требований: функциональные требования, нефункциональные требования.
19. Свойства требований
20. Формализация требований. Цикл работы с требованиями.
21. Введение в технологии разработки программного обеспечения.
22. Понятие процесса разработки ПО.
23. Универсальный процесс.
24. Текущий процесс.

25. Конкретный процесс.
26. Стандартный процесс.
27. Совершенствование
28. процесса.
29. Pull/Push стратегии.
30. Классические модели процесса: водопадная модель, спиральная модель.
31. Фазы и виды деятельности
32. Процессы командной разработки программного обеспечения.
33. Гибкие технологии разработки ПО.
34. Операционная и проектная деятельность.
35. Метод функционального описания предметной области.
36. Нотация структурного функционального моделирования IDEF0.
37. Средства разработки процессных моделей.

Критерии оценивания:

Для каждого вопроса:

- 1 балл дан полный, развёрнутый ответ на поставленный вопрос, изложение материала при ответе – грамотное и логически стройное;
- 0 баллов – обучающийся не владеет материалом по заданному вопросу.

Максимальное количество баллов – 10

Раздел 2. Технология коллективной разработки программного обеспечения

1. Освоение компонентов интерфейса и настройка среды Visual Studio Code предназначенной для обеспечения совместной работы команд разработчиков ПО.
2. Подключение к TFS.
3. Создание командного проекта.
4. Управление жизненным циклом приложений.
5. Архитектурное проектирование.
6. Архитектура и функциональные возможности Visual Studio Code Шаблоны командных проектов.
7. Анализ системных требований для установки Visual Studio Code Анализ процесса установки и конфигурирования TeamFoundationServer2012.

Критерии оценивания:

Для каждого вопроса:

- 1 балл дан полный, развёрнутый ответ на поставленный вопрос, изложение материала при ответе – грамотное и логически стройное;
- 0 баллов – обучающийся не владеет материалом по заданному вопросу.

Максимальное количество баллов – 10

Раздел 3. Управление рисками процесса разработки

1. Управление рисками процесса разработки ПО в программной среде Visual Studio Code
2. Практическая организация и управление рисками процесса разработки ПО в программной среде Visual Studio Code
3. Управление рисками с помощью процесса СММІ
4. Основные понятия риска и риск образующих факторов.
5. Управление рисками на каждом из этапов ЖЦ программного проекта: идентификация рисков, анализ рисков, планирование управления рисками, мониторинг и методы реагирования на риски.
6. Наиболее распространенные риски программных проектов.
7. Методы контроля хода исполнения программных проектов

Критерии оценивания:

Для каждого вопроса:

- 1 балл дан полный, развёрнутый ответ на поставленный вопрос, изложение материала при ответе – грамотное и логически стройное;
- 0 баллов – обучающийся не владеет материалом по заданному вопросу.

Максимальное количество баллов – 10

Раздел 4. Создание и управление проектами с использованием Team Foundation Server

1. Создание проекта
2. Выбор шаблона процесса.
3. Создание группового проекта. .
4. Определение того, какие сценарии должны быть завершены в каждой итерации.
5. Определение нефункциональных требований. Связь нефункциональных требований со сценариями.
6. Разбиение сценариев на истории
7. Определение критериев приемки.
8. Определение требований к отчетам.
9. Создание графика выполнения проекта.
10. Добавление нового разработчика в проект Visual Studio Code
11. Анализ кода при помощи Team Build
12. Создание отчета в Visual Studio 2005 Team Foundation Server
13. создание пользовательской политики возврата после правки в Visual Studio Code Обзор методологий разработки ПО. MSF. IT решение.
14. Основные принципы MSF.
15. Модель команды: основные принципы, ролевые кластеры.
16. Масштабирование команды MSF.
17. Модель процесса.
18. Уровни зрелости процессов по CMMI.
19. Области совершенствования. "Гибкие" (agile) методы разработки. Общее описание "гибких" методов разработки ПО.
20. Extreme Programming: общее описание, основные принципы организации процесса. Scrum: общее описание, роли, практики

Критерии оценивания:

Для каждого вопроса:

- 1 балл дан полный, развёрнутый ответ на поставленный вопрос, изложение материала при ответе – грамотное и логически стройное;
- 0 баллов – обучающийся не владеет материалом по заданному вопросу.

Максимальное количество баллов – 10

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

Процедуры оценивания включают в себя текущий контроль и промежуточную аттестацию.

Текущий контроль успеваемости проводится с использованием оценочных средств, представленных в п. 2 данного приложения. Результаты текущего контроля доводятся до сведения студентов до промежуточной аттестации.

Промежуточная аттестация проводится в форме экзамена

Зачет проводится по окончании обучения до начала экзаменационной сессии. Количество вопросов в задании – 3 (2 теоретических вопроса и 1 практико-ориентированное задание к экзамену). Проверка ответов и объявление результатов производится в день зачета. Результаты аттестации заносятся в зачетную ведомость и зачетную книжку студента. Студенты, не прошедшие промежуточную аттестацию по графику сессии, должны ликвидировать задолженность в установленном порядке.

МЕТОДИЧЕСКИЕ УКАЗАНИЯ ПО ОСВОЕНИЮ ДИСЦИПЛИНЫ

Учебным планом предусмотрены следующие виды занятий:

- практические работы;
- лабораторные работы.

В ходе лабораторных и практических работ развиваются навыки практического применения технологий разработки программного обеспечения для решения задач профессиональной деятельности.

При подготовке к лабораторным занятиям каждый студент должен:

- изучить рекомендованную учебную литературу;

В процессе подготовки к лабораторным и практическим занятиям студенты могут воспользоваться консультациями преподавателя.

Теоретические вопросы должны быть изучены студентами в ходе самостоятельной работы. Контроль самостоятельной работы студентов над учебной программой курса осуществляется методом устного опроса или выполнения практико-ориентированных заданий. В ходе самостоятельной работы каждый студент обязан прочитать основную и по возможности дополнительную литературу по изучаемой теме.

Выделить непонятные термины, найти их значение в литературе.

Для подготовки к занятиям, текущему контролю и промежуточной аттестации студенты могут воспользоваться электронно-библиотечными системами. Также обучающиеся могут взять на дом необходимую литературу на абонементе университетской библиотеки или воспользоваться читальными залами.