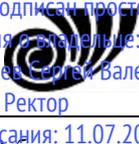


Документ подписан простой электронной подписью Информация о владельце: ФИО: Таскаев Сергей Валерьевич Должность: Ректор Дата подписания: 11.07.2025 06:10:58 Уникальный программный ключ: 054c0182970293149-316996000940392896664	 МИНОБРАЗОВАНИЯ И НАУКИ РОССИИ Федеральное государственное бюджетное образовательное учреждение высшего образования «Челябинский государственный университет» (ФГБОУ ВО «ЧелГУ»)	Рабочая программа дисциплины "Программная инженерия" по направлению подготовки (специальности) 02.03.02 «Фундаментальная информатика и информационные технологии» направленности (профилю) Математические и алгоритмические основы интеллектуальных систем ФГБОУ ВО «ЧелГУ»	стр. 1
---	---	---	--------

Рабочая программа дисциплины (модуля)*

Программная инженерия

Направление подготовки (специальность)

02.03.02 Фундаментальная информатика и информационные технологии

Направленность (профиль)

Математические и алгоритмические основы интеллектуальных систем

Присваиваемая квалификация (степень)

бакалавр

Форма обучения

очная

Год(ы) набора

*Рабочая программа дисциплины (модуля) адаптирована для инклюзивного обучения инвалидов и лиц с ограниченными возможностями здоровья

Челябинск 2025 г.



Содержание

1. Цели освоения дисциплины
2. Место дисциплины в структуре ОПОП
3. Компетенции обучающегося, формируемые в результате освоения дисциплины (модуля)
4. Объем дисциплины (модуля)
5. Структура и содержание дисциплины (модуля)
6. Фонд оценочных средств
 - 6.1. Перечень видов оценочных средств
 - 6.2. Типовые контрольные задания и иные материалы для текущей аттестации
 - 6.3. Типовые контрольные вопросы и задания для промежуточной аттестации
 - 6.4. Критерии оценивания
7. Учебно-методическое и информационное обеспечение дисциплины (модуля)
 - 7.1. Рекомендуемая литература
 - 7.2. Перечень ресурсов информационно-телекоммуникационной сети "Интернет"
 - 7.3. Перечень информационных технологий
8. Материально-техническое обеспечение дисциплины (модуля)
9. Методические указания для обучающихся по освоению дисциплины (модуля)
10. Специальные условия освоения дисциплины обучающимися с инвалидностью и ограниченными возможностями здоровья



1. ЦЕЛИ ОСВОЕНИЯ ДИСЦИПЛИНЫ

Целью данной дисциплины является введение в проблематику, связанную с изучением технологий и средств поддержки жизненного цикла продуктов программного обеспечения и информационных систем, основам экономики и менеджмента создания продуктов ИТ, основам управления качеством продуктов.

Результаты изучения дисциплины направлены на достижение следующих индикаторов:

УК-1.1. Выполняет поиск информации, определяет критерии системного анализа поставленных задач

УК-1.2. Использует критический анализ, систематизацию и обобщение информации для решения поставленных задач

ОПК-2.1. Демонстрирует знание методов использования инструментальных средств, готового программного обеспечения и библиотек; знаком с содержанием Единого реестра российских программ

ОПК-2.2. Демонстрирует умения выбирать и использовать инструментальные средства, готовое программное обеспечение и библиотеки

ОПК-2.3. Имеет практический опыт решения задач анализа, интеграции различных типов программного обеспечения и сетевых коммуникаций

ПК-2.1. Обладает знаниями о методах и средствах сборки модулей и компонент программного обеспечения, о разработке процедур для развертывания программного обеспечения, миграции и преобразования данных, о создании программных интерфейсов; о методах и механизмах оценки и анализа функционирования средств и систем информационных технологий; о международных и профессиональных стандартах информационных технологий, о современных парадигмах и методологиях, инструментальных и вычислительных средствах.

ПК-2.2. Демонстрирует умения: применять методы и средства сборки модулей и компонент программного обеспечения, разработки процедур для развертывания программного обеспечения, миграции и преобразования данных, создания программных интерфейсов; проводить проверку и оценку работоспособности программного продукта.

ПК-2.3. Имеет практический опыт (навыки):

сборки модулей и компонент программного обеспечения, разработки процедур для развертывания программного обеспечения, миграции и преобразования данных, создания программных интерфейсов; оценки работоспособности программного продукта.

ПК-3.1. Обладает знаниями о методах и средствах проектирования программного обеспечения, структур данных, баз данных, программных интерфейсов.

ПК-3.2. Демонстрирует умения: разрабатывать требования к программному продукту, применять методы и средства проектирования программного обеспечения, структур данных, баз данных, программных интерфейсов.

ПК-3.3. Имеет практический опыт (навыки):

проектирования программного обеспечения, структур данных, баз данных, программных интерфейсов.

2. МЕСТО ДИСЦИПЛИНЫ В СТРУКТУРЕ ОПОП

Цикл (раздел) ОПОП: К.М.02.05

2.1 Требования к предварительной подготовке обучающегося:

Технология программирования

Технология баз данных

Управление IT-проектами

Программирование в среде 1С (научный семинар)

Экономика

2.2 Дисциплины и практики, для которых освоение данной дисциплины (модуля) необходимо как предшествующее:

Архитектура ОС Windows (научный семинар)



Преддипломная практика

Подготовка к процедуре защиты и защита выпускной квалификационной работы

Искусственные нейронные сети

3. КОМПЕТЕНЦИИ ОБУЧАЮЩЕГОСЯ, ФОРМИРУЕМЫЕ В РЕЗУЛЬТАТЕ ОСВОЕНИЯ ДИСЦИПЛИНЫ (МОДУЛЯ)

УК-1: Способен осуществлять поиск, критический анализ и синтез информации, применять системный подход для решения поставленных задач

Знать:

Для достижения индикатора УК-1.1:

- методы и средства поиска информации, определяет критерии системного анализа поставленных задач.

Уметь:

Для достижения индикатора УК-1.2:

- использовать критический анализ, систематизацию и обобщение информации для решения поставленных задач.

Владеть:

навыком поиска и обработки информации о процессах производственной деятельности, связанной с созданием и использованием информационных технологий.

ОПК-2: Способен применять компьютерные/суперкомпьютерные методы, современное программное обеспечение, в том числе отечественного происхождения, для решения задач профессиональной деятельности

Знать:

Для достижения индикатора ОПК-2.1:

- методы экономического анализа производственно-хозяйственной деятельности предприятия;
- технологии программной инженерии;
- основные и вспомогательные процессы программной инженерии;
- преимущества инженерного подхода к созданию программного обеспечения;
- основные источники текущей информации по управлению ИТ-сервисами;

Уметь:

Для достижения индикатора ОПК-2.2:

- реализовывать процессы управления качеством производственной деятельности, связанной с созданием и использованием информационных технологий;
- выполнять сопровождение программного обеспечения;
- анализировать предметную область и выделять требования к разрабатываемой программной системе;
- применять теории, методы, алгоритмы, системы и средства информационных технологий при решении профессиональных задач;
- реализовывать проектную деятельность инструментами и методами программной инженерии;

Владеть:

Для достижения индикатора ОПК-2.3:

- навыком управления качеством производственной деятельности, связанной с созданием и использованием информационных технологий.

ПК-2: Способность к осуществлению интеграции программных модулей и компонент и проверки работоспособности программного продукта на основе международных и профессиональных стандартов информационных технологий, современных парадигм и методологий, инструментальных и вычислительных средств, методов и механизмов оценки и анализа функционирования средств и систем информационных технологий

Знать:

Для достижения индикатора ПК-2.1:

- процессы жизненного цикла программного обеспечения.

Уметь:

Для достижения индикатора ПК-2.2:

- разрабатывать и реализовывать процессы жизненного цикла программного обеспечения.



Владеть:

Для достижения индикатора ПК-2.3:
- навыками управления IT-проектами.

ПК-3: Способность к разработке требований и проектированию программного обеспечения на основе применения базовых математических знаний и информационных технологий при решении проектно-технических и прикладных задач

Знать:

Для достижения индикатора ПК-3.1:
- методы мониторинга и оценки качества процессов производственной деятельности, связанной с созданием и использованием информационных технологий;
- методы и средства проектирования программного обеспечения, структур данных, баз данных, программных интерфейсов.

Уметь:

Для достижения индикатора ПК-3.2:
- теории, методы, системы и средства для решения практических задач в области информационных технологий с использованием современных языков, инструментальных средств, сервисов глобальных сетей;
- методы построения математических моделей, используемые при анализе предметной области;
- осуществлять мониторинг и оценку качества процессов производственной деятельности;
- понятие качества ПО, характеристики и атрибуты качества, связь атрибутов качества с требованиями;

Владеть:

Для достижения индикатора ПК-3.3:
- навыками использования методов и механизмов оценки и анализа функционирования средств ИТ.
- навыками реализации проектную деятельность инструментами и методами программной инженерии

В результате освоения дисциплины обучающийся должен

3.1	Знать:
3.1.1	- методы экономического анализа производственно-хозяйственной деятельности предприятия;
3.1.2	- технологии программной инженерии;
3.1.3	- основные и вспомогательные процессы программной инженерии;
3.1.4	- преимущества инженерного подхода к созданию программного обеспечения;
3.1.5	- основные источники текущей информации по управлению ИТ-сервисами;
3.1.6	- методы и средства поиска информации, определяет критерии системного анализа поставленных задач;
3.1.7	- процессы жизненного цикла программного обеспечения;
3.1.8	- методы мониторинга и оценки качества процессов производственной деятельности, связанной с созданием и использованием информационных технологий.
3.2	Уметь:
3.2.1	- использовать критический анализ, систематизацию и обобщение информации для решения поставленных задач;
3.2.2	- реализовывать процессы управления качеством производственной деятельности, связанной с созданием и использованием информационных технологий;
3.2.3	- разрабатывать и реализовывать процессы жизненного цикла программного обеспечения;
3.2.4	- осуществлять мониторинг и оценку качества процессов производственной деятельности;
3.2.5	- реализовывать процессы управления качеством производственной деятельности, связанной с созданием и использованием информационных технологий;
3.2.6	- выполнять сопровождение программного обеспечения;
3.2.7	- анализировать предметную область и выделять требования к разрабатываемой программной системе;
3.2.8	- применять теории, методы, алгоритмы, системы и средства информационных технологий при решении профессиональных задач;
3.2.9	- реализовывать проектную деятельность инструментами и методами программной инженерии;
3.3	Владеть:



Рабочая программа дисциплины "Программная инженерия" по направлению подготовки (специальности)
02.03.02 "Фундаментальная информатика и информационные технологии" направленности (профилю)
Математические и алгоритмические основы интеллектуальных систем ФГБОУ ВО «ЧелГУ»

стр. 6

- | | |
|-------|--|
| 3.3.1 | - поиска и обработки информации о процессах производственной деятельности, связанной с созданием и использованием информационных технологий; |
| 3.3.2 | - управления качеством производственной деятельности, связанной с созданием и использованием информационных технологий; |
| 3.3.3 | - управления IT-проектами; |
| 3.3.4 | - использования методов и механизмов оценки и анализа функционирования средств ИТ. |

4. ОБЪЕМ ДИСЦИПЛИНЫ (МОДУЛЯ)

Общая трудоемкость	4 ЗЕТ
Часов по учебному плану : 144	Виды контроля в семестрах: зачеты с оценкой 7
в том числе :	
аудиторные занятия : 68	
самостоятельная работа : 69,1	
контактная работа: 74,9 ИКР: 6,9	

5. СТРУКТУРА И СОДЕРЖАНИЕ ДИСЦИПЛИНЫ (МОДУЛЯ)

Код занятия	Наименование разделов и тем /вид занятия/	Семестр / Курс	Часов	Литература
Раздел 1. Стандарты разработки ПО				
1.1	Введение в программную инженерию. Модели процесса разработки ПО. Управление проектами. Определения и концепции. Критерии успешности проекта. /Лек/	7	4	Л1.1 Л1.2Л2.1 Л2.2 Э1 Э2
1.2	Программные средства поддержки жизненного цикла ПО CASE-средства – характеристики и технологии. Designer 2000 и Developer 2000 Erwin StarUML /Лаб/	7	8	Л1.1 Л1.2Л2.1 Л2.2 Э1 Э2
1.3	Разработка модели данных в среде StarUML /Ср/	7	12	Л1.1 Л1.2Л2.1 Л2.2 Э1
Раздел 2. Критерии успешности проекта. Организация проектной команды				
2.1	Проект и организационная структура компании. Организация проектной команды. Жизненный цикл проекта /Лек/	7	8	Л1.1 Л1.2Л2.1 Л2.2 Э1 Э2
2.2	BPWin S-designer Case-аналитик StarUML Microsoft PM /Лаб/	7	10	Л1.1 Л1.2Л2.1 Л2.2 Э1 Э2
2.3	Создание диаграмм в среде StarUML /Ср/	7	17,1	Л1.1 Л1.2Л2.1 Л2.2 Э1 Э2
Раздел 3. Базовое расписание проекта. Управление рисками проекта				



3.1	Инициация проекта Базовое расписание проекта. Управление рисками проекта /Лек/	7	10	Л1.1 Л1.2Л2.1 Л2.2 Э1 Э2
3.2	Построение модели предметной области Проектирование интерфейса программы Проектирование структуры данных /Лаб/	7	8	Л1.1 Л1.2Л2.1 Л2.2 Э1 Э2
3.3	Формирование схем зависимости шагов (этапов) проекта. Анализ требований к проекту, возможные методы реализации /Ср/	7	20	Л1.1 Л1.2Л2.1 Л2.2 Э1 Э2
Раздел 4. Рабочее планирование. Принципы качественного управления				
4.1	Оценка трудоемкости и сроков разработки ПО Реализация и сопровождение проекта /Лек/	7	12	Л1.1 Л1.2Л2.1 Л2.2 Э1 Э2
4.2	Проектирование структуры программы (хранимых процедур) Кодирование и отладка программы /Лаб/	7	8	Л1.1 Л1.2Л2.1 Л2.2 Э1 Э2
4.3	Создание отчетов /Ср/	7	20	Л1.1 Л1.2Л2.1 Л2.2 Э1 Э2
Раздел 5. Иная контактная работа				
5.1	Индивидуальные консультации, текущий контроль /ИКР/	7	6,9	Л1.1 Л1.2Л2.1 Л2.2 Э1 Э2 Э3

6. ФОНД ОЦЕНОЧНЫХ СРЕДСТВ

6.1. Перечень видов оценочных средств

Лабораторная работа
Интеллект-карты
Зачет с оценкой

*При дистанционном обучении устный опрос, в том числе защита курсовых работ, реализуется в Microsoft Teams, практические задания, письменные ответы размещаются в Moodle, тестирование осуществляется в Moodle.

6.2. Типовые контрольные задания и иные материалы для текущей аттестации

Самостоятельные домашние работы:

1. Основные понятия программной инженерии.
2. Модели и профили жизненного цикла программных средств.
3. Модели и процессы управления проектами программных средств.
4. Управление требованиями к программному обеспечению.
5. Методы проектирования программных продуктов и признаки их классификации.
6. Языки программирования и их классификация.
7. Тестирование программного обеспечения.

Темы для разработки интеллект-карт:

Сопровождение программного обеспечения.
Конфигурационное управление.
Управление программной инженерией.
Процесс программной инженерии.
Инструменты и методы программной инженерии.
Качество программного обеспечения.
Документирование программного обеспечения.
Технико-экономическое обоснование проектов программных средств.



Задания на лабораторные работы №1-4

1. Задача: инвентаризация оборудования. Процессы: учет компьютерной техники, ее составных частей (клавиатур, мониторов, комплектующих и т.д.) в пределах кафедры (месторасположение, предназначение, ответственный за технику).
2. Задача: поддержание в актуальном состоянии схемы сети. Процессы: учет компьютеров и сетевого оборудования, связей между ними, и истории изменения этих связей.
4. Задача: планирование занятости аудиторий на кафедре. Процессы: ведение реестра аудиторий, характеристики аудиторий, ведение расписания занятий с контролем пересечений.
5. Задача: ведение списка студентов и сотрудников кафедры. Процессы: прием студентов, отчисление, перевод из группы в группу, то же – для сотрудников. Учесть, что студент может быть одновременно сотрудником.
6. Задача: составление учебного плана. Процессы: ведение списка предметов с учетом продолжительности курса, балансировка нагрузки на студентов.
8. Задача: выполнение и контроль лабораторных работ. Процессы: ведение списка студентов, разбиение по подгруппам и бригадам, учет выполнения работы, составления отчета, защиты отчета.
10. Задача: дела студентов. Все документы, касающиеся студента: номер студенческого, зачетка, диплом, заявления, отчеты, курсовые и т.д. Разделение: документ, который содержит оценку и влияет на дальнейшую судьбу студента (оперативный) или просто документ (справочный).
13. Задача: мониторинга торговых точек. Контроль за остатками товара в торговых точках. Мониторинг цен на товар в торговых точках. Контроль за регионально-торговыми представителями заказчика.
14. Задача: Написать программу по моделированию оплаты коммунальных услуг плательщиком. Информационная система должна содержать следующие сведения: фамилия и инициалы плательщика, расчетный счет плательщика, расчетные счета получателя, перечисляемая сумма. Программа должна обеспечивать выбор с помощью меню и выполнение одной из следующих функций: ввод с клавиатуры (из файла) данных о плательщике; вывод на экран информации о сумме, снятой с расчетного счета плательщика, введенного с клавиатуры; вывод на экран информации о плательщиках; осуществление сортировки по разным полям.
15. Задача: студенческая столовая. Процессы: предложить самим.
16. Задача: общежитие.

Лабораторная работа №1. StarUML Этап НАЧАЛО (Inception).

Цель работы – сформировать навыки: работы с реальными заказчиками программных систем; идентификации заинтересованных лиц и интервью с ними;

анализа полученного материала; формулирования проблемы, ее актуальности и потребностей заинтересованных лиц.

Задание. Запустить проект разработки программного обеспечения согласно выданному заданию. Сформировать команду, распределить роли. Выполнить основные действия и создать артефакты, соответствующие данному этапу.

Методические указания и теоретические сведения.

RUP основан на трех ключевых идеях.

Весь ход работ направляется итоговыми целями проекта, выраженными в виде вариантов использования (use cases) — сценариев взаимодействия результирующей программной системы с пользователями или другими системами, при выполнении которых пользователи получают значимые для них результаты и услуги. Разработка начинается с выделения вариантов использования и на каждом шаге контролируется степенью приближения к их реализации.



Основным решением, принимаемым в ходе проекта, является архитектура результирующей программной системы. Архитектура устанавливает набор компонентов, из которых будет построено ПО, ответственность каждого из компонентов (т.е. решаемые им подзадачи в рамках общих задач системы), четко определяет интерфейсы, через которые они могут взаимодействовать, а также способы взаимодействия компонентов друг с другом.

Архитектура является одновременно основой для получения качественного ПО и базой для планирования работ и оценок проекта в терминах времени и ресурсов, необходимых для достижения определенных результатов. Она оформляется в виде набора графических моделей на языке UML.

Основой процесса разработки являются планируемые и управляемые итерации, объем которых (реализуемая в рамках итерации функциональность и набор компонентов) определяется на основе архитектуры.

Этап НАЧАЛО (Inception)

Главное назначение этапа — запустить проект. Основная цель этой фазы — достичь компромисса между всеми заинтересованными лицами относительно задач проекта и выделяемых на него ресурсов. На этой стадии определяются основные цели проекта, руководитель и бюджет, основные средства выполнения — технологии, инструменты, ключевые исполнители. Также, возможно, происходит апробация выбранных технологий, чтобы убедиться в возможности достичь целей с их помощью, и составляются предварительные планы проекта. На эту фазу может уходить около 10% времени и 5% трудоемкости одного цикла.

Цели этапа НАЧАЛО:

определить область применения проектируемой системы (ее предназначение, границы, интерфейсы с внешней средой, критерий признания — приемки);

определить элементы Use Case, критические для системы (основные сценарии поведения, задающие ее функциональность и покрывающие главные проектные решения);

определить общие черты архитектуры, обеспечивающей основные сценарии, создать демонстрационный макет;

определить общую стоимость и план всего проекта и обеспечить детализированные оценки для этапа развития;

идентифицировать основные элементы риска.

Основные действия этапа НАЧАЛО:

формулировка области применения проекта — выявление требований и ограничений, рассматриваемых как критерий признания конечного продукта;

планирование и подготовка бизнес-варианта и альтернатив развития для управления риском, определение персонала, проектного плана, а также выявление зависимостей между стоимостью, планированием и полезностью;

синтезирование предварительной архитектуры, развитие компромиссных решений проектирования; определение решений разработки, покупки и повторного использования, для которых можно оценить стоимость, планирование и ресурсы.

В итоге этапа НАЧАЛО создаются следующие артефакты:

Концепция (Vision) – спецификация представления основных проектных требований, ключевых характеристик и главных ограничений;

начальная модель Use Case (20% от полного представления);

начальный словарь проекта (глоссарий);

начальный бизнес-вариант (содержание бизнеса, критерий успеха — прогноз дохода, прогноз рынка, финансовый прогноз);

начальное оценивание риска;



проектный план, в котором показаны этапы и итерации.

Примеры оформления документов RUP (артефактов) смотри в каталоге «Примеры документов RUP».

Более подробное описание процесса разработки по RUP смотри в документе «ПроцессРазработки_по_RUP.doc».

Рис. 1. Пример хода работ на этапе начала проекта

Лабораторная работа №2. Этап РАЗВИТИЕ (Elaboration).

Задание. Выполнить основные действия и создать артефакты, соответствующие данному этапу разработки программного обеспечения.

Методические указания.

Этап РАЗВИТИЕ (Elaboration)

Главное назначение этапа — создать архитектурный базис системы.

Основная цель этой фазы — на базе основных, наиболее существенных требований разработать стабильную базовую архитектуру продукта, которая позволяет решать поставленные перед системой задачи и в дальнейшем используется как основа разработки системы. На эту фазу может уходить около 30% времени и 20% трудоемкости одного цикла.

Цели этапа РАЗВИТИЕ:

определить оставшиеся требования, функциональные требования формулировать как элементы Use Case;

определить архитектурную платформу системы;

отслеживать риск, устранить источники наибольшего риска;

разработать план итераций этапа КОНСТРУИРОВАНИЕ.

Основные действия этапа РАЗВИТИЕ:

развитие спецификации представления, полное формирование критических элементов Use Case, задающих дальнейшие решения;

развитие архитектуры, выделение ее компонентов.

В итоге этапа РАЗВИТИЕ создаются следующие артефакты:

модель Use Case (80% от полного представления);

дополнительные требования (нефункциональные требования, а также другие требования, которые не связаны с конкретным элементом Use Case);

описание программной архитектуры;

выполняемый архитектурный макет;

пересмотренный список элементов риска и пересмотренный бизнес-вариант;

план разработки для всего проекта, включающий крупноблочный проектный план и показывающий итерации и критерий эволюции для каждой итерации.

Лабораторная работа №3. Этап КОНСТРУИРОВАНИЕ (Construction).

Задание. Выполнить основные действия и создать артефакты, соответствующие данному этапу разработки



программного обеспечения.

Методические указания и теоретические сведения.

Главное назначение этапа — создать программный продукт, который обеспечивает начальные операционные возможности.

Цели этапа **КОНСТРУИРОВАНИЕ**:

минимизировать стоимость разработки путем оптимизации ресурсов и устранения необходимости доработок;

добиться быстрого получения приемлемого качества;

добиться быстрого получения контрольных версий (альфа, бета и т. д.).

Основные действия этапа **КОНСТРУИРОВАНИЕ**:

управление ресурсами, контроль ресурсов, оптимизация процессов;

полная разработка компонентов и их тестирование (по сформулированному критерию эволюции);

оценивание реализаций продукта (по критерию признания из спецификации представления).

В итоге этапа **КОНСТРУИРОВАНИЕ** создаются следующие артефакты:

программный продукт, готовый для передачи в руки конечных пользователей;

описание текущей реализации;

руководство пользователя.

Лабораторная работа №4. Этап **ПЕРЕХОД (Transition)**.

Постановка задачи. Выполнить основные действия и создать артефакты, соответствующие данному этапу разработки программного обеспечения.

Методические указания и теоретические сведения.

Главное назначение этапа — применить программный продукт в среде пользователей и завершить реализацию продукта.

Этап начинается с предъявления пользователям бета-реализации продукта. В ней обнаруживаются ошибки, они корректируются в последующих бета-реализациях. Параллельно решаются вопросы размещения, упаковки и сопровождения продукта. После завершения бета-периода тестирования продукт считается реализованным.

Лабораторная работа №5. Рефакторинг проекта с использованием шаблонов проектирования.

Постановка задачи. Выполнить улучшение разработанного в предыдущих лабораторных работах программного кода. Предусмотреть использование, где это необходимо, шаблонов проектирования.

Лабораторная работа №6. Управление требованиями в среде IBM RequisitePro 7.0.

Постановка задачи. Необходимо оформить требования к программной системе в среде IBM RequisitePro 7.0. [9,10]. Выполнить упражнения по книге Вендрова А.М. [10].

Лабораторная работа №7. Оценка качества программного продукта.

Постановка задачи. С помощью объектно-ориентированных метрик оценить качество программного продукта.

Методические указания и теоретические сведения.



Оценка качества проектирования.

Качество проектирования оценивают с помощью объектно-ориентированных метрик.

Этап РАЗВИТИЕ

Качество логического представления архитектуры оценивают по метрикам:

WMC — взвешенные методы на класс;

NOC — количество детей;

DIT — высота дерева наследования;

NOM — суммарное количество методов, определенных во всех классах системы;

NC — общее количество классов в системе.

Метрики WMC, NOC вычисляются для каждого класса, кроме того, формируются их средние значения в системе. Метрики DIT, NOM, NC вычисляются для всей системы.

Этап КОНСТРУИРОВАНИЕ

На каждой итерации конструирования продукта вычисляются метрики:

WMC — взвешенные методы на класс;

NOC — количество детей;

CBO — сцепление между классами объектов;

RFC — отклик для класса;

LCOM — недостаток связности в методах;

CS — размер класса;

NOO — количество операций, переопределяемых подклассом;

NOA — количество операций, добавленных подклассом;

SI — индекс специализации;

OSavg — средний размер операции;

NPavg — среднее количество параметров на операцию;

NC — общее количество классов в системе;

LOC — суммарная LOC-оценка всех методов системы;

DIT — высота дерева наследования;

NOM — суммарное количество методов в системе.

Метрики WMC, NOC, CBO, RFC, LCOM, CS, NOO, NOA, SI, OSavg, NPavg вычисляются для каждого класса, кроме того, формируются их средние значения в системе. Метрики DIT, NOM, NC, LOCS вычисляются для всей системы.

6.3. Типовые контрольные вопросы и задания для промежуточной аттестации



Вопросы зачетного занятия

1. Предпосылки и история дисциплины. Краткий обзор причин, истории возникновения и становления программной инженерии.
2. Краткая характеристика основных методов проектирования про-грамм, сложившихся в процессе поиска решения основной проблемы программной инженерии - сокращения стоимости программного обеспечения.
3. Процесс создания программ.
4. Показатели качества программного продукта.
5. Стандартизация и стандарты Программной инженерии (SWEBOOK).
6. Понятие жизненного цикла ПО и технологических процессов его раз-работки.
7. Способы организации жизненного цикла ПО, каскадные и итеративные модели жизненного цикла.
8. Набор стандартов, регулирующих процессы разработки ПО в целом.
9. Сравнительные детали различных методологий разработки ПО, предлагаемые в рамках унифицированного процесса разработки Rational (RUP).
10. Экстремальное программирование (XP).
11. Методология MSF.
12. Вопросы, связанные с анализом предметной области и выделением требований к разрабатываемой программной системе.
13. Основные графические модели, используемые при анализе предметной области (диаграммы потоков данных и вариантов использования).
14. Понятие образца проектирования. Классификация образцов проектирования. Примеры образцов анализа и архитектурных стилей.
15. Понятие архитектуры ПО, влияние архитектуры на свойства ПО, а также методы оценки архитектуры.
16. Основные элементы унифицированного языка моделирования UML.
17. Основные факторы удобства использования ПО.
18. Психофизиологические особенности человека, делающие предметы удобными и неудобными для него.
19. Методика проектирования, ориентированная на удобство использования.
20. Тестирование программ. Виды и методы тестирования. Стратегии черного и белого ящика.
21. Статическое и динамическое тестирование. Инспекция кода.
22. Тестовая база, тест, тестовый случай, тестовый монитор, тестовое покрытие и инструменты его измерения.
23. Основные деятельности, входящие в компетенцию руководителей проектов.
24. Аспекты управления ресурсами, персоналом, рисками и коммуникациями проекта.
25. Особенности управления проектами по созданию ПО.
26. Понятие качества ПО, характеристики и атрибуты качества, связь атрибутов качества с требованиями.
27. Краткий обзор различных методов контроля качества ПО, с более детальным рассмотрением тестирования и проверки свойств на моделях.
28. ISO9000: система управления качеством.
29. ISO 12207: процессы качества ПО.
30. CMM: уровни зрелости процессов.

6.4. Критерии оценивания

Зачетное занятие проводится в форме предметной беседы и компьютерного тестирования, рассчитанных на 60 минут.

«Зачтено» – выставляется, если ответ на вопросы и тестирование выполняется студентом на достаточное число баллов и студент не имеет пропусков и задолженности по текущей успеваемости.

Студенты, не набравшие достаточного количества баллов в течение семестра и на зачетной работе и имеющие пропуски занятий, сдают зачет повторно в форме беседы.

«Незачтено» – выставляется студентам, не сдавшим зачет в форме беседы или тестирования.

Эти критерии доводятся до сведения студентов в ходе учебного процесса и проведении консультаций.

В зависимости от уровня подготовленности студента выставляется оценка. При выставлении оценки учитываются:

- результаты текущего контроля;
- посещаемость учебных занятий;
- активность во время занятий;
- участие в научной работе;
- наличие навыков самостоятельной и исследовательской работы.

При необходимости инвалидам и лицам с ограниченными возможностями здоровья предоставляется дополнительное время для подготовки ответа на зачете.

Критерии оценивания:

Характеристики ответа Баллы



Ответ обоснован и полностью соответствует вопросу (заданию). Даются исчерпывающие пояснения. Имеющиеся ошибки незначительны. 30
Ответ соответствует вопросу (заданию), имеются некоторые замечания, но ответ обоснован не в полной мере. 20
Ответ соответствует вопросу (заданию) частично. 10
Ответ не соответствует вопросу (заданию) Менее 10
Лабораторная работа - каждая лабораторная работа - до 20 баллов.
Самостоятельное домашнее задание (по 5 баллов за каждую из работ)

Критерии оценивания зачета:

0-39 баллов – неудовлетворительно

40- 60 - удовлетворительно;

61-80 - хорошо;

81 - 100 - отлично.

При проведении процедуры оценивания результатов обучения инвалидов и лиц с ограниченными возможностями здоровья предусматривается использование технических средств, необходимых им в связи с их индивидуальными особенностями. Эти средства могут быть предоставлены кафедрой или могут использоваться собственные технические средства.

Процедура оценивания результатов обучения инвалидов и лиц с ограниченными возможностями здоровья по дисциплине) предусматривает предоставление информации в формах, адаптированных к ограничениям их здоровья и восприятия информации:

Для лиц с нарушениями зрения:

- в печатной форме увеличенным шрифтом,
- в форме электронного документа,
- в форме аудиофайла,

Для лиц с нарушениями слуха:

- в печатной форме,
- в форме электронного документа.

Для лиц с нарушениями опорно-двигательного аппарата:

- в печатной форме,
- в форме электронного документа,
- в форме аудиофайла.

Данный перечень может быть конкретизирован в зависимости от контингента обучающихся.

При проведении процедуры оценивания результатов обучения инвалидов и лиц с ограниченными возможностями здоровья по дисциплине обеспечивается выполнение следующих дополнительных требований в зависимости от индивидуальных особенностей обучающихся:

- при необходимости инструкция по порядку проведения процедуры оценивания предоставляется в доступной форме (устно, в письменной форме)
- доступная форма предоставления заданий оценочных средств (в печатной форме, в печатной форме увеличенным шрифтом, в форме электронного документа, задания зачитываются преподавателем);
- доступная форма предоставления ответов на задания (письменно на бумаге, набор ответов на компьютере, с использованием услуг ассистента, устно).

При необходимости для обучающихся с ограниченными возможностями здоровья и инвалидов процедура оценивания результатов обучения по дисциплине может проводиться в несколько этапов.

Проведение процедуры оценивания результатов обучения инвалидов и лиц с ограниченными возможностями здоровья допускается с использованием дистанционных образовательных технологий.

7. УЧЕБНО-МЕТОДИЧЕСКОЕ И ИНФОРМАЦИОННОЕ ОБЕСПЕЧЕНИЕ ДИСЦИПЛИНЫ (МОДУЛЯ)

7.1. Рекомендуемая литература

7.1.1. Основная литература

Авторы,	Заглавие	Издательство,	Ресурс
---------	----------	---------------	--------



	Авторы,	Заглавие	Издательство,	Ресурс
Л1.1	Киселева Т. В.	Программная инженерия: курс лекций (https://biblioclub.ru/index.php?page=book&id=563341)	Ставрополь : Северо-Кавказский Федеральный университет (СКФУ), 2018	ЭБС
Л1.2	Лагереv Д. Г., Коростелев Д. А., Азарченков А. А., Коптенок Е. В.	Программная инженерия: лабораторный практикум: практикум (https://biblioclub.ru/index.php?page=book&id=602232)	Москва, Берлин : Директ -Медиа, 2021	ЭБС

7.1.2. Дополнительная литература

	Авторы,	Заглавие	Издательство,	Ресурс
Л2.1	Червенчук И. В., Грицай А. С.	Моделирование объектно ориентированных систем с помощью UML: учебное пособие (https://biblioclub.ru/index.php?page=book&id=682248)	Омск : Омский государственный и технический университет (ОмГТУ), 2019	ЭБС
Л2.2	Крумина К. В., Полковникова С. Г.	Управление проектами: учебное пособие (https://biblioclub.ru/index.php?page=book&id=683233)	Омск : Омский государственный и технический университет (ОмГТУ), 2020	ЭБС

7.2. Перечень ресурсов информационно-телекоммуникационной сети "Интернет"

Э1	Инженерия программного обеспечения [Электронный ресурс] : Введение в программную инженерию // Интернет университет информационных технологий. – Электрон. дан. – Режим доступа: http://www.intuit.ru/studies/higher_education/3406/courses/353/info			
Э2	Инженерия программного обеспечения [Электронный ресурс] : Компонентный подход в программировании // Интернет университет информационных технологий. – Электрон. дан. – Режим доступа: https://www.intuit.ru/studies/higher_education/3406/courses/64/info			
Э3	Лань [Электронный ресурс] : электронно-библиотечная система (ЭБС) / издательство Лань. – URL: http://e.lanbook.com/ .			

7.3 Перечень информационных технологий

7.3.1 Программное обеспечение

Adobe Reader
Microsoft Office Professional Plus 2010 (Лицензия Троицкого филиала)
LMS Moodle
Freemind
GanttProject
Dia
StarUML

7.3.2 Профессиональные базы данных и информационно-справочные системы

1.	eLIBRARY.RU [Электронный ресурс]: электронная библиотека / Науч. электрон. б-ка. – URL: http://elibrary.ru
2.	ИНФОРМИО [Электронный ресурс] : электронный справочник [обеспечение всех типов образовательных учреждений нормативными, методическими, научно-практическими материалами]. – URL: http://www.informio.ru/ .
3.	Национальная электронная библиотека (НЭБ) [Электронный ресурс] : объединенный электронный каталог фондов российских библиотек : сайт. – URL: http://нэб.рф .
4.	Архив научных журналов [Электронный ресурс] : база данных / Национальный электронно-информационный консорциум (НП НЭИКОН). – URL: www.neicon.ru/cons
5.	Консультант Плюс [Электронный ресурс] : справочно-правовая система : база данных



8. МАТЕРИАЛЬНО-ТЕХНИЧЕСКОЕ ОБЕСПЕЧЕНИЕ ДИСЦИПЛИНЫ (МОДУЛЯ)

Аудитории для проведения занятий оснащены интерактивным оборудованием (интерактивная доска, ноутбук, проектор). Есть доступ к методическим материалам, наглядным пособиям, материалам для проведения практических занятий. Имеется свободный доступ в компьютерные классы, доступ в Интернет и WI-FI, для проведения телемостов и интернет-конференций.

Материально-техническое обеспечение реализации рабочей программы дисциплины «Программная инженерия» включает:

- основную и дополнительную литературу;

- учебные аудитории (посадочные места не менее 15) с проекторами мультимедиа-оборудованием (проектор, ноутбук или стационарный компьютер) для проведения занятий в зависимости от занятости аудиторного фонда филиала;

- учебная аудитория № 215 для проведения занятий и самостоятельной работы студентов с неограниченным доступом в Интернет, ЭБС и т.п.;

- наличие помещений для самостоятельной работы с компьютерной техникой и с возможностью подключения к сети "Интернет" и обеспечением доступа в электронную информационно-образовательную среду организации.

Обучение инвалидов и лиц с ограниченными возможностями здоровья осуществляется с применением следующего оборудования: мультимедийный проектор (использование презентаций с укрупненным текстом), ноутбуки.

Освоение дисциплины инвалидами и лицами с ограниченными возможностями здоровья осуществляется с использованием средств обучения общего и специального назначения:

– лекционная аудитория – мультимедийное оборудование; источники питания для индивидуальных технических средств; использование презентаций с укрупненным текстом; колонки для усиления звука;

– учебная аудитория № 215 для практических занятий – мультимедийное оборудование; колонки для усиления звука;

– учебная аудитория № 215 для самостоятельной работы – стандартные рабочие места с персональными компьютерами; рабочее место с персональным компьютером, с программой экранного доступа, и программой экранного увеличения.

В каждой аудитории, где обучаются инвалиды и лица с ограниченными возможностями здоровья, предусмотрено соответствующее количество мест для обучающихся с учетом ограничений их здоровья.

10. Описание наборов демонстрационного оборудования и учебно-методических пособий, обеспечивающих тематические иллюстрации занятий лекционного типа

В аудитории для проведения лекционных занятий имеется демонстрационное оборудование: Проектор Panasonic, Интерактивная доска SMART Technologies SMART Board 680V, графический планшет, ноутбуки (10 шт.) IRU Patriot 505 (i3/2Gb/320Gb/HD 5470 1 Gb/ DVD RW/Wifi/15,6/Cam).

Для осуществления образовательного процесса используются программные продукты: Free Mind, Lazarus, Microsoft Office 2010, Microsoft Windows 7, 7 Zip, Gantt Project, редактор Айрен, StarUML, Adobe Reader. В аудитории имеется свободный доступ в Интернет (Wi-Fi).

Материал для тематической иллюстрации занятий лекционного типа оформлен в виде презентаций.

В аудитории имеется свободный доступ в Интернет (Wi-Fi).

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

Освоение содержания учебной дисциплины «Программная инженерия» осуществляется на лекциях, лабораторных занятиях и в процессе самостоятельной учебной деятельности студентов.

В ходе изучения дисциплины уделяется большое внимание как теоретическому усвоению базовых понятий программной инженерии, так и приобретению, развитию и закреплению компетенций, практических навыков и умений по использованию инструментов и методов программной инженерии, позволяющих проектировать, разрабатывать, тестировать, документировать и внедрять программный продукт.

Лабораторные занятия ориентированы на отработку основных методов работы данными, элементами управления, библиотеками кодов, вспомогательными пакетами программ. В ходе выполнения лабораторных работ используется среда моделирования и проектирования StarUML). Применяются СУБД (MySQL, PostgreSQL и др.). Также



используется сервер приложений (Microsoft, Apache Tomcat). Для структурирования информации применяется программа FreeMindMap позволяющая строить как индивидуальные интеллект-карты, так и адаптированные для интерактивной работы в группах. Полученные в ходе выполнения лабораторных работ результаты обсуждаются как в индивидуальном порядке, так и в дискуссионном формате.

Другим направлением учебной деятельности студентов является самостоятельная работа по предложенным вопросам. Внимательно ознакомьтесь с вопросами, которые предусматривают самостоятельное изучение. Затем следует найти источники информации по соответствующему вопросу, используя предложенный преподавателем список обязательной и дополнительной литературы, а также Интернет-ресурсы. Во время чтения целесообразно осуществлять теоретический анализ текста: выделять главные мысли, находить аргументы, подтверждающие основные тезисы, а также иллюстрирующие их примеры и т.д. После этого можно приступить к выполнению задания (составление конспекта, заполнение таблицы и др.). При этом важно помнить, что выполненное задание во всех случаях должно отражать основные выводы, к которым вы пришли в процессе самостоятельной учебной деятельности.

Если в процессе самостоятельной работы возникают затруднения (непонимание отдельных положений дисциплины, трудности в выполнении заданий и др.), обучающемуся следует обратиться за консультацией к преподавателю, ведущему занятия в соответствующей группе.

В освоении дисциплины инвалидами и лицами с ограниченными возможностями здоровья большое значение имеет индивидуальная работа.

Под индивидуальной работой подразумевается две формы взаимодействия с преподавателем: индивидуальная учебная работа (консультации), т.е. дополнительное разъяснение учебного материала и углубленное изучение материала с теми обучающимися, которые в этом заинтересованы, и индивидуальная воспитательная работа. Индивидуальные консультации по предмету является важным фактором, способствующим индивидуализации обучения и установлению воспитательного контакта между преподавателем и обучающимся инвалидом или обучающимся с ограниченными возможностями здоровья.

Методические указания при дистанционном обучении

В случае применения при обучении дисциплины электронного обучения, дистанционных образовательных технологий общение обучающихся и преподавателя осуществляется в режиме реального времени (Microsoft Teams, форумы, электронная почта, сотовая связь) и отложенного времени (системы дистанционного обучения Moodle, электронная почта, форумы).

Большую часть времени обучающиеся самостоятельно работают с учебно-методическими материалами. Студенты имеют возможность консультироваться с преподавателем по всем вопросам, возникающим в ходе самостоятельной работы посредством системы дистанционного обучения Moodle, электронной почты, сотовой связи, форумов.

Доступ обучающегося к учебным ресурсам в режиме отложенного времени, самостоятельной работы осуществляется через сеть Интернет в удобном для него месте, времени и темпе.

10. СПЕЦИАЛЬНЫЕ УСЛОВИЯ ОСВОЕНИЯ ДИСЦИПЛИНЫ ОБУЧАЮЩИМИСЯ С ИНВАЛИДНОСТЬЮ И ОГРАНИЧЕННЫМИ ВОЗМОЖНОСТЯМИ ЗДОРОВЬЯ

Освоение дисциплины инвалидами и лицами с ограниченными возможностями здоровья осуществляется с использованием специальных технических средств и информационных технологий, предоставляемых Ресурсным учебно-методическим центром по обучению инвалидов и лиц с ограниченными возможностями здоровья ЧелГУ по запросу обучающегося (мобильные специальные технические средства для лиц с нарушениями зрения и с нарушением слуха, ассистивные информационные технологии).

При необходимости для обучающихся с нарушениями зрения на рабочих местах для проведения практических или лабораторных занятий устанавливается специальное программное обеспечение (программа речевой навигации, речевые синтезаторы, экранные лупы).

В учебные аудитории обеспечивается беспрепятственный доступ для обучающихся с инвалидностью и с ограниченными возможностями здоровья. В каждой аудитории, где обучаются инвалиды и лица с ограниченными возможностями здоровья, предусматривается соответствующее количество мест для обучающихся с учетом нарушений их здоровья.

Для освоения дисциплины инвалидам и лицам с ограниченными возможностями здоровья предоставляется доступ к печатным источникам, имеющимся в научной библиотеке ЧелГУ, с помощью специальных технических средств; доступ с помощью специальных технических и программных средств к электронным источникам, представленным в форме электронного документа в фонде научной библиотеки ЧелГУ или электронно-библиотечных системах.

Учебно-методические материалы для обучающихся из числа инвалидов и лиц с ограниченными возможностями здоровья предоставляются в формах, адаптированных к ограничениям их здоровья и особенностям восприятия информации.



МИНОБРНАУКИ РОССИИ
Федеральное государственное бюджетное образовательное
учреждение высшего образования
«Челябинский государственный университет» (ФГБОУ ВО «ЧелГУ»)

Рабочая программа дисциплины "Программная инженерия" по направлению подготовки (специальности)
02.03.02 "Фундаментальная информатика и информационные технологии" направленности (профилю)
Математические и алгоритмические основы интеллектуальных систем ФГБОУ ВО «ЧелГУ»

стр. 18

Для инвалидов и лиц с ограниченными возможностями здоровья освоение дисциплины может быть частично или полностью осуществлено с использованием дистанционных образовательных технологий.

При проведении промежуточной аттестации по дисциплине обучающимся с инвалидностью и с ограниченными возможностями здоровья обеспечивается по их заявлению предоставление в доступной форме в зависимости от их индивидуальных особенностей инструкции о порядке проведения промежуточной аттестации, оценочных средств и возможности ответов на задания (письменно на бумаге, набор ответов на компьютере, письменно шрифтом Брайля, с использованием услуг ассистента, устно).

При проведении процедуры оценивания результатов обучения инвалидов и лиц с ограниченными возможностями здоровья предусматривается использование предоставленных ЧелГУ или собственных технических средств, необходимых им в связи с их индивидуальными особенностями. При необходимости инвалидам и лицам с ограниченными возможностями здоровья предоставляется дополнительное время для подготовки ответа на задания, процедура оценивания результатов обучения по дисциплине может проводиться в несколько этапов.

