



МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ
РОССИЙСКОЙ ФЕДЕРАЦИИ

Федеральное государственное бюджетное образовательное учреждение
высшего образования
«Магнитогорский государственный технический университет им. Г.И. Носова»



УТВЕРЖДАЮ
Директор ИЭиАС
С.И. Лукьянов

26.02.2020 г.

РАБОЧАЯ ПРОГРАММА ДИСЦИПЛИНЫ (МОДУЛЯ)

CALS-ТЕХНОЛОГИИ В РАЗРАБОТКЕ ПРОГРАММНЫХ СРЕДСТВ

Направление подготовки (специальность)

09.04.01 Информатика и вычислительная техника

Направленность (профиль/специализация) программы

Программное обеспечение средств вычислительной техники и автоматизированных систем

Уровень высшего образования - магистратура

Форма обучения

очная

Институт/ факультет	Институт энергетики и автоматизированных систем
Кафедра	Вычислительной техники и программирования
Курс	1
Семестр	2

Магнитогорск
2020 год

Рабочая программа составлена на основе ФГОС ВО - магистратура по направлению подготовки 09.04.01 Информатика и вычислительная техника (приказ Минобрнауки России от 19.09.2017 г. № 918)

Рабочая программа рассмотрена и одобрена на заседании кафедры
Вычислительной техники и программирования

19.02.2020 г. протокол № 5


Зав. кафедрой  О.С. Логунова

Рабочая программа одобрена методической комиссией ИЭ и АС

26.02.2020 г. протокол № 5

Председатель  С.И. Лукьянов

Рабочая программа составлена:

доцент кафедры ВТиП, канд. техн. Наук  А. В. Леднов

Рецензент:

начальник отдела технологических платформ

ООО «Компас Плюс», канд. техн. наук



Д.С. Сафонов

Лист актуализации рабочей программы

Рабочая программа пересмотрена, обсуждена и одобрена для реализации в 2021 - 2022 учебном году на заседании кафедры Вычислительной техники и программирования

Протокол от _____ 20__ г. № ____
Зав. кафедрой _____ О.С. Логунова

Рабочая программа пересмотрена, обсуждена и одобрена для реализации в 2022 - 2023 учебном году на заседании кафедры Вычислительной техники и программирования

Протокол от _____ 20__ г. № ____
Зав. кафедрой _____ О.С. Логунова

1 Цели освоения дисциплины (модуля)

Целью преподавания дисциплины (модуля) «Cals технологии в разработке программных средств» является ознакомление с совокупностью базовых принципов, управленческих и информационных технологий, обеспечивающих поддержку жизненного цикла изделий на всех его стадиях, базирующихся на использовании интегрированной информационной среды, в которой посредством электронного обмена данными реализуется взаимодействие всех участников жизненного цикла: заказчиков продукции, разработчиков, производителей продукции, эксплуатантов.

2 Место дисциплины (модуля) в структуре образовательной программы

Дисциплина CALS-технологии в разработке программных средств входит в часть учебного плана формируемую участниками образовательных отношений образовательной программы.

Для изучения дисциплины необходимы знания (умения, владения), сформированные в результате изучения дисциплин/ практик:

Анализ и описание профессиональной информации

Современные проблемы информатики и вычислительной техники

Знания (умения, владения), полученные при изучении данной дисциплины будут необходимы для изучения дисциплин/практик:

Основы теории машинного обучения

Проектирование и тестирование сложных пользовательских интерфейсов

Промышленные информационные системы

Технология разработки программного обеспечения

Выполнение и защита выпускной квалификационной работы

3 Компетенции обучающегося, формируемые в результате освоения дисциплины (модуля) и планируемые результаты обучения

В результате освоения дисциплины (модуля) «CALS-технологии в разработке программных средств» обучающийся должен обладать следующими компетенциями:

Код индикатора	Индикатор достижения компетенции
ПК-3	Владеет навыками описания информационных и математических моделей, технических решений с точки зрения специалиста по информационным технологиям и математических моделей
ПК-3.1	Оценивает качество математических моделей и технических решений
ПК-7	Обладает способностью к управлению процессом, внутренним правилам, методик и регламентов проведения работ по разработке программного обеспечения
ПК-7.1	Оценивает качество управления проведения работ по разработке программного обеспечения
ПК-9	Владение знаниями и навыками разработки проектной документации по проектированию интерфейсов, созданию методик оценки интерфейсов, концептуальному проектированию интерфейсов и созданию структурных руководств по проектированию интерфейсов
ПК-9.1	Оценивает качество проектирования и разработки сложных интерфейсов программного обеспечения

2.1 Стандарт ISO 10303 (STEP)- Системы автоматизации производства и их интеграция. Представление данных об изделии и обмен этими данными. Стандарт ISO 13584 (PLIB) Системы автоматизации производства и их интеграция. Библиотека деталей.	2	4	4/2И		18	1. Поиск дополнительной информации по заданной теме. 2. Самостоятельное изучение учебной и научно литературы. 3. Работа с электронными библиотеками.	1. Проверка индивидуальных заданий 2. Устный опрос	ПК-3.1, ПК-7.1, ПК-9.1
2.2 Стандарт ISO 15531 (MANDATE). Промышленные автоматизированные системы и интеграция. Данные по управлению промышленным производством. Стандарт ISO 8879 (SGML). Обработка информации — текст и офисные системы — стандартный обобщённый язык разметки. XML		5	5/2И		19,05	1. Поиск дополнительной информации по заданной теме. 2. Самостоятельное изучение учебной и научно литературы. 3. Работа с электронными библиотеками.	1. Проверка индивидуальных заданий 2. Устный опрос	ПК-3.1, ПК-7.1, ПК-9.1
Итого по разделу		9	9/4И		37,05			
Итого за семестр		17	17/8И		73,05		зачёт	
Итого по дисциплине		17	17/8И		73,05		зачет	

5 Образовательные технологии

1. Поиск дополнительной информации по заданной теме.
2. Самостоятельное изучение учебной и научно литературы.
3. Работа с электронными библиотеками.

1. Традиционные образовательные технологии, ориентированные на организацию образовательного процесса и предполагающую прямую трансляцию знаний от преподавателя к аспиранту.

Формы учебных занятий с использованием традиционных технологий:

Информационная лекция – последовательное изложение материала в дисциплинарной логике, осуществляемое преимущественно вербальными средствами (монолог преподавателя).

Лабораторная работа – организация учебной работы с реальными материальными и информационными объектами, экспериментальная работа с аналоговыми моделями реальных объектов.

2. Технологии проблемного обучения – организация образовательного процесса, которая предполагает постановку проблемных вопросов, создание учебных проблемных ситуаций для стимулирования активной познавательной деятельности аспирантов.

3. Интерактивные технологии – организация образовательного процесса, которая предполагает активное и нелинейное взаимодействие всех участников, достижение на этой основе лично значимого для них образовательного результата.

Формы учебных занятий с использованием специализированных интерактивных технологий:

Лекция «обратной связи» – лекция–провокация (изложение материала с заранее запланированными ошибками), лекция–беседа, лекция–дискуссия, лекция–конференция.

4. Информационно-коммуникационные образовательные технологии – организация образовательного процесса, основанная на применении программных сред и технических средств работы со знаниями в различных предметных областях.

6 Учебно-методическое обеспечение самостоятельной работы обучающихся

Представлено в приложении 1.

7 Оценочные средства для проведения промежуточной аттестации

Представлены в приложении 2.

8 Учебно-методическое и информационное обеспечение дисциплины (модуля)

а) Основная литература:

1/ Оганян, К. М. Объектно- и субъектно-ориентированные CASE-технологии в социальной работе : учебное пособие / К. М. Оганян, К. К. Оганян. - Москва : Инфра-М : Znanium.com, 2015. - 156 с. - ISBN 978-5-16-103596-2. - Текст : электронный. - URL: <https://znanium.com/catalog/product/522023> (дата обращения: 28.10.2020). – Режим доступа:

б) Дополнительная литература:

1. Масляев В. С. Управление жизненным циклом программных систем. методические указания к выполнению самостоятельной и лабораторных работ– Томск : Изд-во Томск. гос. ун-та систем управления и радиоэлектроники, 2016. – 13 с. [Электронный ресурс]: научно-образовательный портал ТУСУРа. – URL: http://aoi.tusur.ru/upload/methodical_materials/Methodicheskie_ukazaniya_k_vypolneniju_LR_U_ZHCPS_file__702_9479.pdf— Загл. с экрана

в) Методические указания:

1. Масляев В. С. Управление жизненным циклом программных систем. методические указания самостоятельной и лабораторных работ– Томск : Изд-во Томск. гос. ун-та систем управления и радиотехники, 2013. – 13 с. [Электронный ресурс]: научно-образовательный портал ТУСУР http://aoi.tusur.ru/upload/methodical_materials/Methodicheskie_ukazaniya_k_vypolneniju_LR_UZHSCPS_f
Загл. с экрана

г) Программное обеспечение и Интернет-ресурсы:

Программное обеспечение

Наименование ПО	№ договора	Срок действия
MS Windows 7 Professional(для классов)	Д-1227-18 от 08.10.2018	11.01.2019
MS Office 2007 Professional	№ 135 от 17.09.2007	без срока действия
MS Office Visio Prof 2003(для классов)	Д-1227-18 от 08.10.2018	11.01.2019
MS Visual Studio Code	свободно распространяемое ПО	без срока действия

Профессиональные базы данных и информационные справочные системы

Название курса	Срок действия

9 Материально-техническое обеспечение дисциплины (модуля)

Материально-техническое обеспечение дисциплины включает:

1. Лекционная аудитория ауд. 282. Мультимедийные средства хранения, передачи и представления информации.
2. Компьютерные классы Центра информационных технологий ФГБОУ ВО «МГТУ имени Г.И. Носовского», объединенные в локальные сети с выходом в Internet, оснащенные современными компьютерами, методическими комплексами для решения задач в области информатики и вычислительной техники.
3. Аудитории для самостоятельной работы: компьютерные классы; читальные залы библиотек УИТ и АСУ с персональными компьютерами, выходом в Интернет и с доступом в электронную образовательную среду университета.
4. Аудиторий для групповых и индивидуальных консультаций, текущего контроля успеваемости и аттестации. Ауд. 282 и классы УИТ и АСУ.

«Учебно-методическое обеспечение самостоятельной работы обучающихся»

CALS-ТЕХНОЛОГИИ В РАЗРАБОТКЕ ПРОГРАММНЫХ СРЕДСТВ

Упражнение 1

Первоначальная настройка git. Инициализация каталога. Состояния файлов в git. Первый коммит.

Цель занятия – провести первоначальную настройку системы контроля версий git, после установки инициализировать каталог для работы, разобраться с существующими состояниями файлов в git, сделать первый коммит.

Шаг 1: Зайти в папку {Номер группы} и в ней создать папку соответствующую инициалам студента на английском языке. Например, для студента Иванов Петр Петрович, папка будет иметь имя IPP.

Шаг 2: Провести инициализацию репозитория в созданной папке. Для этого, открыть программу Git Bash, перейти в созданную папку (для перемещения используется команда `cd T://{Номер группы}/{Инициалы}`)

Шаг 3: Установить настройки имени и e-mail'a, не используя опцию `--global`.

Шаг 4: Создать в папке файл `my_first_file.txt` и проиндексировать его.

Шаг 5: Сделать первый коммит.

Шаг 6: Открыть файл `my_first_file.txt` и добавить в него строчку "test row". Проиндексировать изменения.

Шаг 7: Создать новый файл `my_second_file.txt`. Проиндексировать изменения.

Шаг 8: Сделать второй коммит.

Упражнение 2

Игнорирование файлов. Сравнение изменений. Удаление и перемещение файлов.

Цель занятия – научиться исключать файлы, которые нет необходимости вести в системе контроля версий. Получить практические навыки сравнения проделанных изменений в файлах.

Шаг 1: Создать папку `temp` в своем репозитории.

Шаг 1: Создать папку `log` и добавить в нее 2 файла: `main.html` и `some.tmp`.

Шаг 1: Создать файл `.gitignore` и добавить в игнорирование папку `temp` и файлы с расширением `.tmp` из папки `log`.

Шаг 1: Закоммитить добавление файла `.gitignore`.

Шаг 1: Внести изменения в файл `my_first_file.txt`, добавив строчку "row to index", проиндексировать данные изменения. Еще раз внести изменения в файл, добавив строчку "row no index".

Шаг 1: Посмотреть индексированные и неиндексированные изменения используя команду `git diff`.

Шаг 1: Удалить файл `my_first_file.txt`, зафиксировать данное удаление.

Шаг 1: Переименовать файл `my_second_file.txt` в `my_first_file.txt`, зафиксировать изменение.

Упражнение 3

Отмена внесенных изменений. Работа с метками.

Цель занятия – научиться отменять сделанные изменения, работать с метками.

Шаг 1: Создать три файла: `1.txt`, `2.txt`, `3.txt`.

Шаг 2: Проиндексировать первый файл и сделать коммит с комментарием "add 1.txt file".

- Шаг 3: Проиндексировать второй и третий файлы.
Шаг 4: Удалить из индекса второй файл.
Шаг 6: Перезаписать уже сделанный коммит с новым комментарием “add 1.txt and 3.txt”
Шаг 7: Создать аннотированную метку с названием v0.01.
Шаг 8: Создать легковесную ветку указывающую на первый коммит в репозитории.

Упражнение 4

Работа с ветками, решение конфликтов

Цель занятия – научиться создавать ветки, перемещаться по ним, объединять и удалять их. Решать конфликты слияния

Шаг 1: Создать новую ветку my_first_branch.

Шаг 2: Перейти на ветку и создать новый файл in_branch.txt, закоммитить изменения.

Шаг 3: Вернуться на ветку master.

Шаг 4: Создать и сразу перейти на ветку new_branch.

Шаг 5: Сделать изменения в файле 1.txt, добавить строчку “new row in 1.txt file”, закоммитить изменения.

Шаг 6: Перейти на ветку master и слить ветки master и my_first_branch, после чего слить ветки master и new_branch.

Шаг 7: Удалить ветки my_first_branch и new_branch.

Шаг 8: Создать ветки branch_1 и branch_2.

Шаг 9: Перейти на ветку branch_1 и изменить файл 1.txt, удалить все содержимое и добавить текст “fix in 1.txt”, изменить файл 3.txt, удалить все содержимое и добавить текст “fix in 3.txt”, закоммитить изменения.

Шаг 10: Перейти на ветку branch_2 и также изменить файл 1.txt, удалить все содержимое и добавить текст “My fix in 1.txt”, изменить файл 3.txt, удалить все содержимое и добавить текст “My fix in 3.txt”, закоммитить изменения.

Шаг 11: Слить изменения ветки branch_2 в ветку branch_1.

Шаг 12: Решить конфликт файла 1.txt в ручном режиме, а конфликт 3.txt используя команду git mergetool с утилитой Meld.

Упражнение 5

Создание документа XML

Шаг 1: Проработайте ниже предложенный пример создания XML-документа и его отображения с помощью каскадных таблиц стилей.

Шаг 2: Создайте в текстовом редакторе Notepad новый файл и введите текст XML-документа, сохранив с расширением .xml

```

<?xml version="1.0" encoding="windows-1251"?>
<!--Имя файла:fale_1.xml-->
<FILE_1>
  <STUDENT>
    <FAMILY>Иванов</FAMILY>
    <NAME>Сергей </NAME>
    <YEAR>1993</YEAR>
    <GROUP>ИФ 87</GROUP>
  </STUDENT>
  <STUDENT>
    <FAMILY>Петрова</FAMILY>
    <NAME>Галина </NAME>
    <YEAR>1992</YEAR>
    <GROUP>ИФ 87</GROUP>
  </STUDENT>
  <STUDENT>
    <FAMILY>Семенов</FAMILY>
    <NAME>Валерий </NAME>
    <YEAR>1993</YEAR>
    <GROUP>ИФ 88</GROUP>
  </STUDENT>
  <STUDENT>
    <FAMILY>Павлова</FAMILY>
    <NAME>Ирина </NAME>
    <YEAR>1994</YEAR>
    <GROUP>ИФ 89</GROUP>
  </STUDENT>
</FILE_1>

```

Данный документ состоит из двух основных частей: пролога и корневого документа (называемого также элементом документа). Элемент документа называется здесь FILE_1, его начальный тег - <FILE_1>, а конечный - </FILE_1>, а содержимое - 4 вложенных элемента STUDENT. В свою очередь каждый элемент STUDENT содержит ряд вложенных элементов.

Шаг 3: Откройте документ с помощью браузера Internet Explorer. После проверки синтаксиса, документ отобразится на экране. При наличии ошибок вместо документа на экран будет выдано сообщение о невозможности отобразить страницу.

Шаг 4: Попробуйте изменить степень детализации представления элементов документа. Щелкните на символе знака минус (-) слева от начального тега, чтобы свернуть элемент, либо на знаке плюс (+) рядом со свернутым элементом, чтобы развернуть его. Например, щелкнув на знаке минус (-) рядом с элементом FILE_1, вы получите то же, что представлено на рисунке:

```

<?xml version="1.0" encoding="windows-1251" ?>
<!-- Имя файла:fale_1.xml -->
+ <FILE_1>

```

Шаг 5: Создайте в файле file_2.css каскадную таблицу стилей:

```

STUDENT
    {display:block;
    margin-top: 12pt;
    font-size: 10 pt}
FAMILY
    {font-style:italic}
NAME
    {font-weight:bold}

```

Шаг 6: Откройте в текстовом редакторе файл, созданный в первом пункте задания, и второй строкой документа следующую инструкцию по обработке:

```
<?xml version="1.0" encoding="windows-1251"?>
<?xml-stylesheet type="text/css" href="file_2.css"?>
<!--Имя файла:fale_10.xml-->
<FILE_1>
  <STUDENT>
```

Шаг 7: Создайте XML- документ, представляющий информацию по определенной вариантом предметной области. Созданный документ должен соответствовать следующим требованиям:

1. документы должны иметь глубину вложенности не менее четырех элементов;
2. число элементов документа, не имеющих вложенных, должно быть не менее пяти;
3. элементы документа должны содержать комментарии о своем содержании;
4. документ должен включать элементы, содержащие символьные данные и дочерний элементы;

Шаг 8: Создайте таблицу каскадных стилей, которая отформатирует созданный XML- документ. Созданная CSS-таблица должна соответствовать следующим правилам:

1. CSS-таблица должна включать как контекстуальные, так и родовые селекторы;
2. дочерние элементы должны наследовать CSS-формат родительского элемента;
3. созданная CSS-таблица должна импортировать другую таблицу стилей;
4. таблица стилей должна включать использование атрибута STYLE;

Шаг 8: В отчете представить код xml-документа, код таблиц CSSи скриншот табличного представления документа.

Варианты предметных областей создаваемых XML-документов:

1. библиографическое описание списка литературы
2. список студентов факультета
3. список изучаемых дисциплин

«Оценочные средства для проведения промежуточной аттестации»

CALS-ТЕХНОЛОГИИ В РАЗРАБОТКЕ ПРОГРАММНЫХ СРЕДСТВ

Код индикатора	Индикатор достижения компетенции	Оценочные средства
ПК-3 Владеет навыками описания информационных и математических моделей, технических решений с точки зрения специалиста по информационным технологиям и математических моделей		
Код	<i>Содержание индикатора</i>	<i>Теоретические вопросы, тесты, практические задания, задачи из профессиональной области, комплексные задания, в том числе задания на курсовые проекты (работы) или иные материалы, оценивающие индикатор достижения компетенции</i>
ПК-3.1	Оценивает качество математических моделей и технических решений	<ol style="list-style-type: none"> 1. Что такое Единое информационное пространство Жизненного цикла изделия? 2. Какие информационные технологии служат для реализации CALS и каким образом? 3. Провести первоначальную настройку системы контроля версии git, после установки инициализировать каталог для работы, разобраться с существующими состояниями файлов в git, сделать первый коммит. 4. Научиться исключать файлы, которые нет необходимости вести в системе контроля версий. Получить практические навыки сравнения проделанных изменений в файлах.
ПК-7 Обладает способностью к управлению процессом, внутренних правил, методик и регламентов проведения работ по разработке программного обеспечения		
Код	<i>Содержание индикатора</i>	<i>Теоретические вопросы, тесты, практические задания, задачи из профессиональной области, комплексные задания, в том числе задания на курсовые проекты (работы) или иные материалы, оценивающие индикатор достижения компетенции</i>
ПК-7.1	Оценивает качество управления проведения работ по разработке программного обеспечения	<ol style="list-style-type: none"> 1. Обзор системы контроля версий Git. 2. Сравнение систем контроля версий Git и SVN. 3. Работа с ветками, решение конфликтов. Цель работы: научиться создавать ветки, перемещаться по ним, объединять и удалять их. Решать конфликты слияния.

Код индикатора	Индикатор достижения компетенции	Оценочные средства
ПК-9 Владение знаниями и навыками разработки проектной документации по проектированию интерфейсов, созданию методик оценки интерфейсов, концептуальному проектированию интерфейсов и созданию структурных руководств по проектированию интерфейсов		
Код	<i>Содержание индикатора</i>	<i>Теоретические вопросы, тесты, практические задания, задачи из профессиональной области, комплексные задания, в том числе задания на курсовые проекты (работы) или иные материалы, оценивающие индикатор достижения компетенции</i>
ПК-9.1	Оценивает качество проектирования и разработки сложных интерфейсов программного обеспечения	1. Просмотр истории коммитов, команда git log. Цель работы: освоить механизм работы с командой git log для получения информации об истории коммитов. 2. Обзор платформы github.com 3. Работа с удаленным репозиторием. Github.com. Цель работы: научиться работать с удаленным репозиторием, использовать платформу github.com