

МИНОБРНАУКИ РОССИИ
ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ
ВЫСШЕГО ОБРАЗОВАНИЯ
«ВОРОНЕЖСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ»
(ФГБОУ ВО «ВГУ»)

УТВЕРЖДАЮ

Заведующий кафедрой
электроники



Усков Г.К.

20.05.2025 г.

РАБОЧАЯ ПРОГРАММА УЧЕБНОЙ ДИСЦИПЛИНЫ

Б1.В.03.ДВ.01.01 Коллективная среда разработки программного обеспечения

1. Код и наименование направления подготовки/специальности:

09.04.01 Информатика и вычислительная техника

2. Профиль подготовки/специализация:

Автоматизированные информационно-измерительные системы

3. Квалификация выпускника: магистр

4. Форма обучения: очная

5. Кафедра, отвечающая за реализацию дисциплины: электроники

6. Составители программы:

Овчинникова Татьяна Михайловна, к.ф.-м.н., доцент

7. Рекомендована: НМС физического факультета 20.05.2025, № протокола: 5

8. Учебный год: 2026/2027

Семестр(ы)/Триместр(ы): 3

9. Цели и задачи учебной дисциплины

Целью — выработка базовых знаний в области создания, сопровождения и поддержки программных проектов, навыков коллективной разработки программного обеспечения на основе современных методов и стандартов систем контроля версий, разработки, отладки и дальнейшего развития программных продуктов.

Задачи:

- дать возможность студентам овладеть на практике методами управления системами контроля версий;
- дать возможность студентам овладеть на практике методами управления системами отслеживания ошибок, средства автоматизации тестирования;
- дать возможность студентам овладеть на практике методами управления системами управления проектами, их применением при коллективной работе над проектами в области ИТ.

10. Место учебной дисциплины в структуре ООП:

Дисциплина относится к вариативной части блока Б1.

11. Планируемые результаты обучения по дисциплине/модулю (знания, умения, навыки), соотнесенные с планируемыми результатами освоения образовательной программы (компетенциями) и индикаторами их достижения:

Код	Название компетенции	Код(ы)	Индикатор(ы)	Планируемые результаты обучения
ПК-3	Способен управлять программно-техническими и технологическими ресурсами для разработки компьютерного программного обеспечения в области автоматизированных информационных-измерительных систем	ПК-3.1	Владеть методами управления инфраструктурой коллективной среды разработки компьютерного программного обеспечения в области автоматизированных информационно-измерительных систем	Владеть: методами управления инфраструктурой коллективной среды разработки компьютерного программного обеспечения в области автоматизированных информационно-измерительных систем.

12. Объем дисциплины в зачетных единицах/час – 2 / 72

Форма промежуточной аттестации *зачет*

13. Трудоемкость по видам учебной работы

Вид учебной работы		Трудоемкость	
		Всего	По семестрам
			1
Аудиторные занятия		34	34
в том числе:	лекции	34	34
	практические		
	лабораторные		
Самостоятельная работа		38	38
Форма промежуточной аттестации: зачет.			
Итого:		72	72

13.1. Содержание дисциплины

№ п/п	Наименование раздела дисциплины	Содержание раздела дисциплины	Реализация раздела дисциплины с помощью
-------	---------------------------------	-------------------------------	---

			онлайн-курса, ЭУМК*
1. Лекции			
1.1	Особенности коллективной разработки программного обеспечения	Понятие коллективной разработки программного обеспечения. Основные типы программных средств поддержки процесса коллективной разработки ПО. Особенности совместной работы над программными проектами. Среда разработки ПО и их основные функции, обеспечивающие возможность коллективной разработки ПО	–
1.2	Системы контроля версий	Обзор. Терминология. Общие принципы работы. Локальные, централизованные и распределенные системы контроля версий. Обзор рынка систем контроля версий. Mercurial. Subversion. Git	–
1.3	Apache Subversion	История создания. Структура хранилища данных. Дельта-кодирование. Рабочая копия. Протоколы доступа к хранилищу. Правки: создание, фиксация, смешивание. Справочная система. Идентификация правок: числовые идентификаторы, ключевые слова, даты. Создание рабочей копии. Внесение изменений в рабочую копию. Анализ изменений. Обновление рабочей копии. Публикация изменений рабочей копии в хранилище. Решение конфликтов. Ветвление. Использование веток. Создание ветки. Работа с веткой. Копирование изменений между ветками. Копирование отдельных изменений. Слияние веток. Основные операции. Создание репозитория, рабочей копии, разрешение конфликтов обновления. Ветвление в Subversion. Разрешение конфликтов слияния. Коллективная разработка ПО с применением Subversion.	–
1.4	Mercurial	История создания. Основы работы в Mercurial. Репозиторий. Децентрализация. Ветвление. Ревизии. Коммиты. Наборы изменений. Вершины. Теги. Слияния. Возможности.	–
1.5	Git	История создания. Основы работы в Git. Ветвление, как основа системы контроля версий, достоинства и недостатки. Слелки. Области хранения файлов. Создание и клонирование репозитория. Игнорирование файлов. Жизненный цикл файлов в Git. Работа с удаленными репозиториями. Основные операции. Создание и настройка локального репозитория, разрешение конфликтов обновления. Ветвление в Git. Разрешение конфликтов слияния. Коллективная разработка ПО с применением Git	–
1.6	Системы отслеживания ошибок, средства автоматизации тестирования	Отслеживание ошибок, как неотъемлемая часть разработки и сопровождения программного обеспечения. Классификация программных ошибок. Основные атрибуты отчета об ошибке. Жизненный цикл ошибки. Рынок систем отслеживания ошибок. Интеграция в среды разработки ПО. Системы отслеживания ошибок. Особенности применения систем контроля ошибок при коллективной разработке программного обеспечения. Системы отслеживания ошибок на примере GitLab / GitHub / GitVerse. Рассмотрение особенностей работы коллектива разработчиков и тестировщиков при разработке ПО. Подходы по автоматизации процесса отслеживания ошибок	–
1.7	Системы управления проектами, их применение при коллективной работе над проектами в области ИТ	Основные виды архитектуры программного обеспечения. Особенности проектирования ПО и применения архитектурных шаблонов. Основные функции систем управления проектами. Преимущества использования систем управления проектами. Функции менеджера проекта. Слежение за проектом. Типы организационных структур при работе над проектами. Обзор систем управления проектами. Шаблоны проектирования архитектуры программного обеспечения. Системы управления проектами при коллективной разработке программного обеспечения. Система управления проектами.	–
2. Практические занятия			

3. Лабораторные занятия

13.2. Темы (разделы) дисциплины и виды занятий

№ п/п	Наименование темы (раздела) дисциплины	Виды занятий (количество часов)				
		Лекции	Практические	Лабораторные	Самостоятельная работа	Всего
1	Особенности коллективной разработки программного обеспечения	4			8	12
2	Системы контроля версий	5			5	10
3	Apache Subversion	5			5	10
4	Mercurial	5			5	10
5	Git	5			5	10
6	Системы отслеживания ошибок, средства автоматизации тестирования	5			5	10
7	Системы управления проектами, их применение при коллективной работе над проектами в области ИТ	5			5	10
Итого:		34			38	72

14. Методические указания для обучающихся по освоению дисциплины:

Для успешного освоения дисциплины необходима регулярная и планомерная работа с конспектом лекций и литературой.

После окончания лекции нужно просматривать конспект для определения материала, вызывающего затруднения для понимания. После этого следует обратиться к рекомендуемой в настоящей программе литературе с целью углубленного изучения проблемного вопроса. В общем случае работа лишь с одним литературным источником часто является недостаточной для полного понимания, поэтому необходимо просматривать несколько источников для выбора того, который наиболее полно и доступно освещает изучаемый материал. В случае если проблемы с пониманием остались, необходимо обратиться к преподавателю на ближайшей лекции с заранее сформулированными вопросами.

Для успешного освоения лекционного курса рекомендуется регулярно повторять изученный материал.

Самостоятельная работа студентов способствует глубокому усвоению учебного материала и развитию навыков самообразования. Самостоятельная работа может включать в себя следующие составляющие:

- работа с текстами: учебниками, справочниками, дополнительной литературой, а также проработка конспектов лекций;
- работа над темами для самостоятельного изучения;
- участие в работе студенческих научных конференций, олимпиад;
- подготовка зачету.

Кроме литературы из основного списка рекомендуется самостоятельно использовать дополнительную. Независимо от вида учебника, работа с ним должна происходить в течение всего семестра. Эффективнее работать с учебником не после, а перед лекцией.

В процессе самостоятельной работы следует занимать активную позицию и пользоваться не только рекомендованной литературой, но и самостоятельно найденными источниками. При изучении дисциплины рекомендуется использовать возможности сети Интернет для получения дополнительной информации по рассматриваемой теме.

При использовании дистанционных образовательных технологий и электронного обучения выполнять все указания преподавателей, вовремя подключаться к онлайн-занятиям, ответственно подходить к выполнению заданий для самостоятельной работы.

15. Перечень основной и дополнительной литературы, ресурсов интернет, необходимых для освоения дисциплины

а) основная литература:

№ п/п	Источник
1.	Липаев В. В. Программная инженерия сложных заказных программных продуктов : учебное пособие. — Москва: МАКС Пресс, 2014, 309 с.
2.	Зубкова Т. М. Технология разработки программного обеспечения : учебное пособие. — Оренбург: Оренбургский государственный университет, ЭБС АСВ, 2017. — 469 с.

б) дополнительная литература:

№ п/п	Источник
1.	Бесплатный учебник по Git и GitHub [Электронный ресурс]. — URL: https://htmlacademy.ru/blog/html_old
2.	Учебные руководства по Jira: изучите принципы Agile с помощью Jira [Электронный ресурс]. — URL: https://www.atlassian.com/ru/agile/tutorials
3.	Руководство по использованию Mercurial [Электронный ресурс]. — URL: https://wiki.mercurial-scm.org/RussianTutorial
4.	Управление версиями в Subversion [Электронный ресурс]. — URL: https://svnbook.red-bean.com/index.ru.html
5.	Понуторай П. Git: контроль версий. 3-е изд / П. Понуторай, Д. Лолигер. — Астана: «Спринт Бук», 2025. — 512 с.: ил. — ISBN 978-601-08-4358-5
6.	Кейнер Кем Тестирование программного обеспечения: контекстно ориентированный подход / Кем Кейнер, Джеймс Бах, Брет Петтикорт. — СПб.: Питер, 2025. — 352 с.: ил. — (Серия «Библиотека программиста»). — ISBN 978-5-4461-2165-6

в) информационные электронно-образовательные ресурсы (официальные ресурсы интернет)*:

№ п/п	Ресурс
1.	Электронная библиотека Зональной научной библиотеки Воронежского госуниверситета : электронно-библиотечная система. — URL : https://lib.vsu.ru/zgate?lnit+elib.xml,simple_elib.xsl+rus
2.	Электронно-библиотечная система "БиблиоТех" : электронно-библиотечная система. — URL : https://lib.vsu.ru/?p=4&t=2d&id=1486
3.	Электронно-библиотечная система «ЮРАЙТ» : электронно-библиотечная система. — URL : https://lib.vsu.ru/?p=4&t=2d&id=1457
4.	Электронно-библиотечная система BOOK.ru.(изд-во "КноРус") : электронно-библиотечная система. — URL : https://lib.vsu.ru/?p=4&t=2d&id=1436
5.	Национальный цифровой ресурс "РУКОНТ" : электронно-библиотечная система. — URL : https://lib.vsu.ru/?p=4&t=2d&id=1401
6.	Электронно-библиотечная система "ZNANIUM.COM" (изд-во "ИНФРА-М") : электронно-библиотечная система. — URL : https://lib.vsu.ru/?p=4&t=2d&id=1360
7.	Электронно-библиотечная система ibook.ru : электронно-библиотечная система. — URL : https://lib.vsu.ru/?p=4&t=2d&id=1344
8.	Электронно-библиотечная система IPRbooks : электронно-библиотечная система. — URL : https://lib.vsu.ru/?p=4&t=2d&id=1343
9.	Электронно-библиотечная система «КнигаФонд» : электронно-библиотечная система. — URL : https://lib.vsu.ru/?p=4&t=2d&id=1336
10.	Электронно-библиотечная система IQLib : электронно-библиотечная система. — URL : https://lib.vsu.ru/?p=4&t=2d&id=1310
11.	Электронно-библиотечная система "Издательство "Лань" : электронно-библиотечная система. — URL : https://lib.vsu.ru/?p=4&t=2d&id=1308
12.	Электронно-библиотечная система "Университетская библиотека online" : электронно-библиотечная система. — URL : https://lib.vsu.ru/?p=4&t=2d&id=1307
13.	Электронно-библиотечная система "Консультант студента" : электронно-библиотечная система. — URL : https://lib.vsu.ru/?p=4&t=2d&id=1306

16. Перечень учебно-методического обеспечения для самостоятельной работы

№ п/п	Источник
1.	Зубра А. С. Культура умственного труда студента : пособие для студентов вузов / А.С.Зубра. — 2-е изд., испр. и доп. — Мн. : Дикта, 2007. — 228с.
2.	Горцевский А.А. Организация самостоятельной работы студента / А.А. Горцевский, М.И. Любицына. — Л. : ЛГУ, 1958. — 50 с.

17. Образовательные технологии, используемые при реализации учебной дисциплины, включая дистанционные образовательные технологии (ДОТ, электронное обучение (ЭО), смешанное обучение):

При реализации дисциплины для проведения текущего контроля и в качестве информационного ресурса используются технологии электронного обучения и

дистанционные образовательные технологии на базе образовательного портала "Электронный университет ВГУ" по адресу edu.vsu.ru, а также другие доступные ресурсы сети Интернет.

18. Материально-техническое обеспечение дисциплины:

Мультимедийная аудитория (ауд. 401): специализированная мебель, компьютеры, мультимедиа-проектор, настенный экран для проектора, аудио колонки

WinPro 8, Linux Debian, Open Office, Google Chrome

Помещение для самостоятельной работы обучающихся (ауд. 401): специализированная мебель, мультимедиа-проектор, настенный экран для проектора, аудио колонки, компьютеры с возможностью подключения к сети «Интернет» и обеспечением доступа в электронную информационно-образовательную среду ВГУ

WinPro 8, Linux Debian, Open Office, Google Chrome, Visual Studio Code, StarUML, Maxima, Octave, MATLAB, JVM, Scala, Haskel, Closure, Java, Kotlin, Python, Go, GCC, CLANG, ReactiveX, VHDL, Verilog, ReactiveX, VHDL, Verilog, SimulIDE Circuit Simulator, Wokwi Simulator, NI LabView, Arduino Studio, MicroCap Evaluation

19. Оценочные средства для проведения текущей и промежуточной аттестаций

Порядок оценки освоения обучающимися учебного материала определяется содержанием следующих разделов дисциплины:

№ п/п	Наименование раздела дисциплины (модуля)	Компетенция(и)	Индикатор(ы) достижения компетенции	Оценочные средства
1.	Особенности коллективной разработки программного обеспечения Системы контроля версий Apache Subversion Mercurial Git Системы отслеживания ошибок, средства автоматизации тестирования Системы управления проектами, их применение при коллективной работе над проектами в области ИТ	ПК-3 Способен управлять программно-техническими и технологическими ресурсами для разработки компьютерного программного обеспечения в области автоматизированных информационно-измерительных систем	ПК-3.1 Владеть методами управления инфраструктурой коллективной среды разработки компьютерного программного обеспечения в области автоматизированных информационно-измерительных систем	Устный опрос
Промежуточная аттестация форма контроля – зачет				Перечень теоретических вопросов

20. Типовые оценочные средства и методические материалы, определяющие процедуры оценивания

Оценка знаний, умений и навыков, характеризующая этапы формирования компетенций в рамках изучения дисциплины осуществляется в ходе текущей и промежуточной аттестаций.

Текущая аттестация проводится в соответствии с Положением о текущей аттестации обучающихся по программам высшего образования Воронежского государственного университета. Текущая аттестация проводится в формах выполнения практико-ориентированных заданий – устного опроса на основе case-заданий.

Промежуточная аттестация проводится в соответствии с Положением о промежуточной аттестации обучающихся по программам высшего образования.

20.1. Текущий контроль успеваемости

Контроль успеваемости по дисциплине осуществляется с помощью следующих оценочных средств: устного опроса на основе case-заданий. Пример case-задания приведен ниже:

Список вопросов из собеседования на позицию DevOps:

1. Расскажите о работе в команде в общем
2. Расскажите об особенностях работы в команде и организации команды при разработке ПО
3. Расскажите о нескольких командах IT проекта и их взаимодействие
4. Расскажите о ролях в команде(разработчик, тимлид, техлид, проджект менеджер, системный аналитик, бизнес аналитик, владелец проекта, тестировщицу)
5. Расскажите о видах организации проектов (водопадная, итерационная, плюсы мину-сы, когда используются)
6. Расскажите о видах версий (альфа бетта, пререлиз, релиз)
7. Расскажите о методологии Agile
8. Расскажите о методологии Scrum
9. Расскажите о том, что такое баг трекары и таск менеджеры
10. Расскажите о том, что такое Система контроля версий
11. Расскажите о том, что такое Continuous Integration
12. Расскажите о том, что такое Вики системы
13. Расскажите о том, что такое Системы контроля соглашений о коде (например sonar)
14. Расскажите о том, что такое автоматическое тестирование (junit и подобные, htmlunit и подобные, jMeter и подобные)
15. Расскажите о том, что такое ручное тестирование
16. Расскажите о составление технической документации
17. Расскажите о правилах создания задач в таск менеджере
18. Расскажите о подходе "клиент-сервер" или "разделение труда" при организации команды. Плюсы и минусы, как избежать минусов и усилить плюсы

Описание технологии проведения.

Текущая аттестация проводится в соответствии с Положением о текущей аттестации обучающихся по программам высшего образования Воронежского государственного университета – в форме устного опроса на основе case-заданий. Критерии оценивания приведены ниже. Опрос выполняется на лекционном занятии при рассмотрении реальных ситуаций из области профессиональной деятельности.

Результаты текущей аттестации учитываются преподавателем при проведении промежуточной аттестации (зачета).

Критерии оценки компетенций (результатов обучения) при выполнении теста:

Критерии оценивания компетенций	Уровень сформированности компетенций	Шкала оценок
80–100% правильных ответов	Повышенный уровень	Отлично
66–79% правильных ответов	Базовый уровень	Хорошо
50–65% правильных ответов	Пороговый уровень	Удовлетворительно
0–49% правильных ответов	–	Неудовлетворительно

20.2. Промежуточная аттестация

Промежуточная аттестация по дисциплине осуществляется в форме зачета. Согласно П ВГУ 2.1.07 – 2024 Положению о проведении промежуточной аттестации обучающихся по образовательным программам высшего образования, оценка на зачете может быть выставлена по результатам текущей успеваемости обучающегося в течение семестра и на основании процедуры и критериев оценивания, представленных в рабочей программе, но не ранее чем на заключительном занятии.

Промежуточная аттестация по дисциплине (зачет) осуществляется с помощью следующих оценочных средств: теоретических вопросов. В контрольно-измерительный материал включаются два теоретических вопроса, позволяющих оценить уровень полученных знаний, умений и навыков.

Перечень вопросов к зачету:

1. Понятие коллективной разработки программного обеспечения.
2. Основные типы программных средств поддержки процесса коллективной разработки ПО.
3. Особенности совместной работы над программными проектами.
4. Среды разработки ПО и их основные функции, обеспечивающие возможность коллективной разработки ПО
5. Система контроля версий. Обзор. Терминология.
6. Система контроля версий. Общие принципы работы. Локальные, централизованные и распределенные системы контроля версий.
7. Обзор рынка систем контроля версий. Mercurial. Subversion. Git
8. Subversion. История создания.
9. Subversion. Структура хранилища данных. Дельта-кодирование.
10. Subversion. Рабочая копия. Протоколы доступа к хранилищу.
11. Subversion. Правки: создание, фиксация, смешивание. Справочная система.
12. Subversion. Идентификация правок: числовые идентификаторы, ключевые слова, даты.
13. Subversion. Создание рабочей копии. Внесение изменений в рабочую копию.
14. Subversion. Анализ изменений. Обновление рабочей копии. Публикация изменений рабочей копии в хранилище.
15. Subversion. Решение конфликтов.
16. Subversion. Ветвление. Использование веток. Создание ветки. Работа с веткой. Копирование изменений между ветками. Копирование отдельных изменений. Слияние веток.
17. История создания. Основы работы в Mercurial.
18. Mercurial. Репозиторий. Децентрализация.
19. Mercurial. Ветвление. Ревизии. Коммиты.
20. Mercurial. Наборы изменений. Вершины.
21. Mercurial. Теги. Слияния. Возможности.
22. Git. История создания. Основы работы в Git.
23. Git. Ветвление, как основа системы контроля версий, достоинства и недостатки.
24. Git. Слепки.
25. Git. Области хранения файлов.
26. Git. Создание и клонирование репозитория. Игнорирование файлов.
27. Git. Жизненный цикл файлов в Git.
28. Git. Работа с удаленными репозиториями.
29. Git. Создание и настройка локального репозитория, разрешение конфликтов обновления.
30. Git. Коллективная разработка ПО с применением Git

31. Отслеживание ошибок, как неотъемлемая часть разработки и сопровождения программного обеспечения.
32. Классификация программных ошибок. Основные атрибуты отчета об ошибке. Жизненный цикл ошибки.
33. Рынок систем отслеживания ошибок.
34. Интеграция в среды разработки ПО. Системы отслеживания ошибок.
35. Особенности применения систем контроля ошибок при коллективной разработке программного обеспечения.
36. Системы отслеживания ошибок на примере GitLab / GitHub / GitVerse. Рассмотрение особенностей работы коллектива разработчиков и тестировщиков при разработке ПО.
37. Подходы по автоматизации процесса отслеживания ошибок в GitLab / GitHub / GitVerse.
38. Основные виды архитектуры программного обеспечения.
39. Особенности проектирования ПО и применения архитектурных шаблонов.
40. Основные функции систем управления проектами.
41. Преимущества использования систем управления проектами.
42. Функции менеджера проекта. Слежение за проектом.
43. Типы организационных структур при работе над проектами.
44. Обзор систем управления проектами.
45. Шаблоны проектирования архитектуры программного обеспечения.
46. Системы управления проектами при коллективной разработке программного обеспечения.

Описание технологии проведения

Промежуточная аттестация проводится в соответствии с Положением о промежуточной аттестации обучающихся по программам высшего образования.

Требования к выполнению заданий (шкалы и критерии оценивания).

Для оценивания результатов обучения на зачете используются следующие **показатели:**

- 1) Владеть на практике методами управления системами контроля версий;
- 2) Владеть на практике методами управления системами отслеживания ошибок, средства автоматизации тестирования;
- 3) Владеть на практике методами управления системами управления проектами, их применением при коллективной работе над проектами в области ИТ.

Для оценивания результатов обучения на зачете используется **шкала:** «зачтено», «не зачтено».

Соотношение показателей, критериев и шкалы оценивания результатов обучения:

Критерии оценивания компетенций	Уровень сформированности компетенций	Шкала оценок
Обучающийся владеет частично теоретическими основами дисциплины, фрагментарно способен к использованию типовых подходов и средств реализации практических задач.	Базовый уровень	Зачет
Обучающийся демонстрирует отрывочные, фрагментарные знания, допускает грубые ошибки.	–	Не зачтено

ЛИСТ СОГЛАСОВАНИЙ**РАБОЧАЯ ПРОГРАММА УЧЕБНОЙ ДИСЦИПЛИНЫ****Направление/специальность**

09.04.01 Информатика и вычислительная техника

Дисциплина

Б1.В.03.ДВ.01.01 Коллективная среда разработки программного обеспечения

Профиль подготовки/специализация

Автоматизированные информационно-измерительные системы

Форма обучения очная**Учебный год** 2025/2026

Ответственный исполнитель

Доцент кафедры электроники _____ .__ 20__

СОГЛАСОВАНО

Куратор ООП

по направлению/специальности _____ .__ 20__

Начальник отдела обслуживания ЗНБ _____ .__ 20__

Программа рекомендована НМС физического факультета 28.10.2024 протокол № 8 от 28.10.2024 г.