

МИНИСТЕРСТВО ТРАНСПОРТА РОССИЙСКОЙ ФЕДЕРАЦИИ

**Федеральное государственное автономное образовательное учреждение
высшего образования
«РОССИЙСКИЙ УНИВЕРСИТЕТ ТРАНСПОРТА (МИИТ)»
РУТ (МИИТ)**

Институт транспортной техники и систем управления

Центр «Высшая школа педагогического мастерства»

УТВЕРЖДАЮ

Первый проректор РУТ (МИИТ)

_____ В.С. Тимонин
" ____ " _____ 20__ г.

**ДОПОЛНИТЕЛЬНАЯ ПРОФЕССИОНАЛЬНАЯ ПРОГРАММА
повышения квалификации**

**«Применение системы контроля версий в процессе обучения
информационным технологиям»**
(по направлению подготовки – 27.03.04 «Управление в технических
системах»)

Москва 2022

ЦЕЛЕВАЯ УСТАНОВКА

Дополнительная профессиональная программа «**Применение системы контроля версий в процессе обучения информационным технологиям**» (далее – программа) по повышению квалификации составлена в соответствии с требованиями приказа Минобрнауки России от 01.07.2013 № 499 «Об утверждении Порядка организации и осуществления образовательной деятельности по дополнительным профессиональным программам» с учетом современных потребностей образовательных учреждений высшего образования в дополнительном профессиональном образовании преподавателей, в чью компетенцию входят вопросы преподавания дисциплин математической и инженерной направленности.

Содержание программы соответствует нормам Трудового кодекса Российской Федерации, нормативных актов Российской Федерации и локальных актов, государственных ведомств по направлению образовательной деятельности.

При составлении программы учитывались квалификационные требования к должности преподавателя, указанные в квалификационном справочнике должностей руководителей, специалистов и других служащих, утвержденном постановлением Минтруда России от 21.08.1998 № 37 (в ред. от 12.02.2014).

При разработке программы использовался федеральный государственный образовательный стандарт высшего образования по направлению подготовки 27.03.04 Управление в технических системах (уровень бакалавриата), утвержденный приказом Минобрнауки России от 31.07.2020 № 871.

Цель обучения: совершенствование компетенций при работе в команде, обновление теоретических и практических навыков преподавания дисциплин, связанных с информационными технологиями и проектной деятельностью, в связи с повышением требований к уровню профессиональных знаний и необходимостью освоения современных технологий.

Категории слушателей: лица, имеющие высшее образование.

Форма обучения: заочная.

Трудоемкость программы: 16 часов (16 часов – заочное с применением электронного обучения).

Продолжительность обучения: 1 неделя.

Режим занятий: 2-4 часа в день.

ПЛАНИРУЕМЫЕ РЕЗУЛЬТАТЫ ОБУЧЕНИЯ

Назначением программы повышения квалификации является формирование и укрепление профессиональных компетенций обучаемых в сфере преподавания математических и инженерных дисциплин:

№ п/п	Код и название компетенции	Ожидаемые результаты
1.	ОПК-6. Способен разрабатывать и использовать алгоритмы и программы, современные информационные технологии, методы и средства контроля, диагностики и управления, пригодные для практического применения в сфере своей профессиональной деятельности	Знать: – популярные инструментальные средства для работы с системой контроля версий; – популярные веб-хостинги для работы с системой контроля версий.
		Уметь: – работать с применением системы контроля версий; – применять в учебном процессе системы контроля версий;
		Владеть: – навыками установки необходимого программного обеспечения для работы с применением системы контроля версий; – необходимым объемом навыков и умений для обучения различных категорий обучаемых практике работы с применением системы контроля версий.
2	ОПК-11. Способен понимать принципы работы современных информационных технологий и использовать их для решения задач профессиональной деятельности	Знать: – роль и место системы контроля версий; – основные принципы работы системы контроля версий.
		Уметь: – пользоваться аналогичными инструментальными средствами для работы с применением системы контроля версий.
		Владеть: – терминологией, используемой при работе с применением системы контроля версий; – навыками работы в коллективе с применением системы контроля версий.
	ПК-11. Способен выявлять возможности и потребности приложения программных и	Знать основные методики и технологии для преподавания дисциплин, требующих работы с программными приложениями.

	аппаратных средств автоматизации и управления в системах автоматического управления на транспорте	Уметь применять новые методики и технологии при проведении различных видов занятий.
		Владеть навыками комплексного и внедрения передового опыта преподавания дисциплин, связанных с программными приложениями.

Реализация профессиональных компетенций осуществляется посредством решения частных задач, к которым относятся:

- формирование четкого представления о версионировании продуктов программных средств;
- изучение специальных программных инструментов, необходимых для свободной ориентации и практической реализации применения систем контроля версий;
- совершенствование практических навыков использования системы контроля версий в профессиональной деятельности;
- обновление знаний в контексте практических методик, используемых при работе с применением систем контроля версий.

В результате изучения курса слушатели должны:

- освоить современные средства работы с системой контроля версий;
- применять работу с системой контроля версий в преподавании дисциплин, связанных с информационными технологиями и проектной деятельностью, при работе в команде при решении научно-исследовательских задач.

УЧЕБНЫЙ ПЛАН

№ п/п	Наименование дисциплин и тем	Трудоемкост ь, час.	В том числе		Форм а аттест ации
			Лекции (ДОТ)	Практичес-кие и семинарс-кие занятия, лабораторные работы (ДОТ)	
1.	Введение в системы контроля версий	2	2	-	
2.	Подготовка окружения для работы в команде	2	1	1	
3.	Создание локального репозитория в программе Git Extensions	2	1	1	
4.	Создание и настройка удаленного репозитория	2	1	1	
5.	Рабочий процесс (workflow)	2	1	1	
6.	Клонирование репозитория	2	1	1	
7.	Обзор других сервисов для работы с удаленным репозиторием	2	2	-	
8.	Итоговая аттестация	2	-	2	Зачет
	Итого	16	7	7	16

КАЛЕНДАРНЫЙ УЧЕБНЫЙ ГРАФИК

№ п/п	Наименование дисциплины	Количество учебных часов по учебным неделям (Н) и дням (Д)					Итого
		Заочное (электронное) обучение					
		Д1	Д2	Д3	Д4	Д5	
1	Введение в системы контроля версий	2					2
2	Подготовка окружения для работы в команде		2				2
3	Создание локального репозитория в программе Git Extensions		2				2
4	Создание и настройка удаленного репозитория			2			2
5	Рабочий процесс (workflow)			2			2
6	Клонирование репозитория				2		2
7	Обзор других сервисов для работы с удаленным репозиторием				2		2
8	Итоговая аттестация					2	2
	Всего учебных часов	2	4	4	4	2	16

РАБОЧИЕ ПРОГРАММЫ ДИСЦИПЛИН

Применение системы контроля версий в процессе обучения информационным технологиям

Тема 1 Введение в системы контроля версий

1. Лекция: Актуальность работы с системой контроля версий. Обзор систем контроля версий. Основные понятия (репозиторий, удаленный и локальный). Обзор сервисов для работы над проектами, использующих распределенную систему контроля версий.

Практика: –

Тема 2 Подготовка окружения для работы в команде

Лекция: 2. Подготовка окружения для работы в команде. Установка и настройка необходимых программ (Notepad++, Git, KDiff3, Git Extensions). Регистрация на сервисах для работы с удалёнными репозиториями (GitHub, GitLab, GitFlic). Необходимость установки программ на английском языке.

Практика: Установка и настройка программного обеспечения для работы с системой контроля версий.

Тема 3 Создание локального репозитория в программе Git Extensions

Лекция: 3. Создание локального репозитория в программе Git Extensions. Настройка отображения скрытых файлов и папок. Обзор программы Git Extensions. Философия системы контроля версий (ветки, коммиты, слияние, названия веток). Необходимость файла .gitignore. Настройка файла .gitignore.

Практика: Создание файла .gitignore.

Тема 4 Создание и настройка удаленного репозитория

Лекция: 4. Создание и настройка удаленного репозитория. Настройка удаленного репозитория на локальном компьютере. Первый push. Понятия issue (проблема), pull request (запрос на слияние), collaborator (сотрудник). Добавление файла Readme. Знакомство с языком разметки Markdown.

Практика: Создание и настройка удаленного репозитория.

Тема 5 Рабочий процесс (workflow)

Лекция: 5 Рабочий процесс (workflow). Алгоритм работы при минимальном рабочем процессе:

1. Повтор действий для первых изменений (Initial commit). (создать локальный репозиторий, настроить файл .gitignore, создать удаленный репозиторий, связать удаленный и локальный репозиторий).

2. Создание нового issue. Создание меток. Добавление метки к текущей проблеме.

3. Создание новой ветки в программе Git Extensions. Правила именования веток. Работа в текущей ветке. Одновременная работа с несколькими ветками.

4. Работа над текущей проблемой. Постоянные коммиты в локальный репозиторий (в конце каждого дня обязательно push всех изменений в удаленный репозиторий).

5. После окончания работы над текущей проблемой commit & push все изменения в удаленный репозиторий.

6. Создание pull request. Linked pull request.

7. Назначение reviewer.

8. Итеративная работа по текущему pull request. Понятия Approve, Changes requested. Работа именно в той ветке, которая относится к текущей проблеме. После исправления ВСЕХ замечаний reviewer, commit & push всех изменений. Перезапрос на рецензию.

9. Работа в удаленном репозитории после подтверждения запроса на слияние. Слияние текущей ветки с главной веткой (Merge pull request). Удаление текущей ветки.

10. Работа в локальном репозитории (программа Git Extensions) после слияния ветки – обновление локальной главной ветки.

Практика: Работа с преподавателем по алгоритму минимального рабочего процесса.

Тема 6 Клонирование репозитория

Лекция: 6. Последовательность действий при клонировании удаленного репозитория.

Практика: Клонирование удаленного репозитория.

Тема 7 Обзор других сервисов для работы с удаленным репозиториум

Лекция: 7. Обзор других сервисов для работы с удаленным репозиториум.

Практика: Работа с применением системы контроля версий в IDE Visual Studio, версии старше 2019, веб-хостинга Repl.it.

Итоговая аттестация

Оценка уровня освоения программы слушателями. Анализ качества обучения.

ОРГАНИЗАЦИОННО-ПЕДАГОГИЧЕСКИЕ УСЛОВИЯ

Реализация учебной программы проходит в соответствии с требованиями законодательства Российской Федерации в области образования, нормативными правовыми актами, регламентирующими данные направления деятельности.

При обучении применяются различные виды занятий, при этом используются нижеуказанные обучающие технические комплексы, программы и иные средства, способствующие лучшему теоретическому и практическому усвоению программного материала:

№№ п/п	Наименование технического средства обучения, программного продукта	Количество технических средств обучения и программных продуктов	Количество мест для слушателей	Год выпуска	Примечание
1	2	3	4	5	6
1	Технические комплексы (средства)	-	-	-	-
1.1	Процессор: Intel(R)Core(TM) i3-10100 CPU @ 3.60GHz Оперативная память: 16 ГБ Дисковое устройство: Samsung SSD 970 EVO Plus 500GB Адаптер: Intel(R) UHD Graphics 630, графическая память - 8210 МБ	120	120	2021	-
1.2	Монитор: DELL S2721DGF 165Гц, разрешение - 2560*1440, разрядность - 8 бит, цветовое пространство – SDR	120	120	2021	-
2.	Мультимедийное оборудование	-	-	-	-
2.1	Проектор Acer P5330W:	1	120	2020	-
3.	Обучающие и тестирующие программы	-	-	-	-
3.1	Веб-браузер (Chrome Microsoft Windows: 103.0.5060.53)	120	120	2022	-
3.2	Git, KDiff3, Git Extensions, Notepad++	120	120		
4.	Расходные материалы	-	-	-	-
4.1	Бумага нелинованная А4	400 л	120	2022	-
4.2	Бумага линованная А4	100 л.	120	2022	-

Основные методические материалы (лекции, общие и частные методики) размещаются на электронном носителе и в последующем выдаются слушателям.

Электронный вариант лекций составляют основу теоретической подготовки слушателей, в них систематизированы научные знания по всему

изучаемому курсу, и акцентировано внимание на наиболее сложных и узловых вопросах рассматриваемых тем.

Учебный материал, предусмотренный программой, реализуется на практических занятиях, при выполнении лабораторных работ и в процессе самостоятельной работы обучающихся.

На практических занятиях закрепляются полученные знания лекционного материала и отрабатываются практические умения и навыки с применением системы контроля версий Git и сервиса GitHub. Практические занятия строятся на сочетании индивидуальной и коллективной отработки заданий под руководством преподавателя с самостоятельным принятием экспертных решений. При проведении занятий особое место занимают активные формы и методы обучения: дискуссии; решение задач; анализ существующих решений.

Особое внимание в ходе проведения занятий уделяется внедрению в учебный процесс применением системы контроля версий.

УЧЕБНО-МЕТОДИЧЕСКОЕ ОБЕСПЕЧЕНИЕ РАЗДАТОЧНЫЕ МАТЕРИАЛЫ

- 1) Учебно-методическое издание.
- 2) Тестовые вопросы.

ФОРМЫ АТТЕСТАЦИИ

Итоговая аттестация проводится путем объективной и независимой оценки качества подготовки слушателей. К итоговой аттестации допускаются слушатели, освоившие учебный план в полном объеме.

Форма итоговой аттестации – зачет по результатам тестирования.

ОЦЕНОЧНЫЕ МАТЕРИАЛЫ ТЕСТЫ ДЛЯ ИТОГОВОЙ АТТЕСТАЦИИ

Теория и история государства и права

Система проверки знаний: каждому вопросу соответствует только один правильный ответ. Выполнение задания оценивается следующим образом:

За каждый правильный ответ дается один балл. От 80 % и более – отлично, от 60 % до 80 % – хорошо, от 40 % до 60 % – удовлетворительно, до 40% – неудовлетворительно.

При неудовлетворительном результате слушателю дается две попытки пересдачи. Время на зачет – 0,4 часа/чел.

Тест

1. Технически главная ветка репозитория может иметь название:
Требуется ответ. Один вариант.

(Баллов: 5)

- **любое название**
- только master
- только main
- могут быть две ветки с названиями main и master

2. Для чего нужен Git?

Требуется ответ. Один вариант.

(Баллов: 5)

- **Git – это система управления версиями.**
- Git – это среда разработки.
- Git – это система управления базами данных.
- Git – это новая операционная система.

3. Какая программа для работы с системой контроля версий должна обязательно быть установлена на локальном компьютере?

Требуется ответ. Один вариант.

(Баллов: 6)

- **Git**
- Git Extensions
- KDif3
- Notepad+++

4. Зачем нужен файл .gitignore?

Требуется ответ. Один вариант.

(Баллов: 8)

- **для фильтрации временных и системных файлов, создаваемых операционной системой и средами разработки**
- для фильтрации файлов и папок программной среды, в которой предполагается работать, например *.docx, *.sql
- для фильтрации всех изменений, сделанных в репозитории
- для фильтрации ошибочных версий кода

5. Что такое remote repository?

Требуется ответ. Один вариант.

(Баллов: 5)

- **Удаленный (внешний) репозиторий**
- Локальный репозиторий
- Рабочая ветка
- Пустой репозиторий

6. Что такое `issue`?

Требуется ответ. Один вариант.

(Баллов: 5)

- Задание
- Метка
- Рецензент
- Создается, чтобы потом можно было создать `pull request`

7. Зачем нужен `pull request`?

Требуется ответ. Один вариант.

(Баллов: 8)

- В большинстве случаев `pull request` используется для интеграции нового функционала или для исправления ошибок (**bug**) в основной ветке проекта.
- Обычно `pull request` используется для вытягивания новых изменений, сделанных участниками команды
- В большинстве случаев `pull request` используется для удаления ненужных веток
- Обычно `pull request` используется для закрытия `issue`

8. Что такое слияние (`merge`)?

Требуется ответ. Один вариант.

(Баллов: 8)

- Для согласования изменений, сделанных в управляемой версии коллекции файлов
- Для фиксации всех последних изменений
- Для исправления всех изменений, когда файл изменяется в двух независимых ветках
- Для удаления рабочей ветки и возможности иметь только главную ветку

9. Когда необходимо пригласить рецензента (`reviewer`)?

Требуется ответ. Один вариант.

(Баллов: 8)

- При создании `pull request`
- При закрытии `pull request`
- После `merge` рабочей ветки в главную ветку
- При назначении `issue`

10. Для чего нужно пригласить рецензента (`reviewer`)?

Требуется ответ. Один вариант.

(Баллов: 8)

- Для проверки нового функционала, привнесенного в код
- Для возможности закрыть `issue`
- Для того, чтобы сообщить о `merge`
- Для возможности открыть `pull request`

11. Что делает команда зафиксировать (commit)?

Требуется ответ. Один вариант.

(Баллов: 5)

- Это операция, которая отправляет последние изменения исходного кода в репозиторий, делая эти изменения частью основной ревизии репозитория
- Это операция, которая отправляет последние изменения исходного кода в удаленный репозиторий, делая эти изменения частью основной ревизии репозитория.
- Это операция, которая отправляет все изменения исходного кода в репозиторий, изменяя предыдущие файлы в репозитории

12. Для чего используется команда отправить (push)?

Требуется ответ. Один вариант.

(Баллов: 5)

- **Используется для выгрузки содержимого локального репозитория в удаленный репозиторий**
- Используется для вытягивания содержимого удаленного репозитория в локальный репозиторий
- Используется для слияния рабочей ветки и главной ветки удаленного репозитория
- Используется для фиксации всех новых изменений в локальном репозитории

13. Какие варианты ответа на pull request можно получить от reviewer?

Требуется ответ. Несколько вариантов выбора.

(Баллов: 6)

- **Approve**
- **Request changed**
- **Comment**
- Commit
- Merge

14. При каких вариантах можно создать запрос на слияние (pull request)?

Требуется ответ. Один вариант.

(Баллов: 6)

- **Имеется две ветки (главная и рабочая). Последние изменения были сделаны в рабочей ветке**
- Имеется только главная ветка. Все изменения были push в главную ветку.
- Имеется две главных ветки.

15. Зачем нужно создавать разные ветки?

Требуется ответ. Один вариант.

(Баллов: 6)

- **Работа ведется в рабочей ветке. Всегда есть рабочая версия кода в главной ветке**
- Работа ведется в главной ветке. Другие ветки нужны для того, чтобы сохранять рабочие версии кода, соответствующие заведенным issue
- Работа всегда ведется в глобальном репозитории. Ветки нужно создавать для возможности что-то поправить при обнаружении ошибки

16. После какого ответа рецензента (reviewer) можно вливать (merge) ветку в главную ветку (master)?

Требуется ответ. Один вариант.

(Баллов: 6)

- **Approve**
- Request changed
- Comment
- Commit
- Merge

СПИСОК ЛИТЕРАТУРЫ

№.№ п/п	Наименование	Темы программы
1	Основная литература.	
1.1	Чакон С., Страуб Б. Git для профессионального программиста. СПб.: Питер, 2016. 496 с	
1.2	Васильева М.А., Филиппченко К.М. Система контроля версий. Основы командной разработки : учебное пособие для вузов. Лань, 2022. 144 с.	1-7
1.3	Васильева М.А. ЦИФРОВОЕ ОБРАЗОВАНИЕ КАК ОТВЕТ ВЫЗОВАМ СЕГОДНЯШНЕГО ДНЯ // КОМПЛЕКСНОЕ ВЗАИМОДЕЙСТВИЕ ЛИНГВИСТИЧЕСКИХ И ВЫПУСКАЮЩИХ КАФЕДР В ТЕХНИЧЕСКОМ ВУЗЕ Международная научно-практическая конференция посвященная 125-летию РУТ (МИИТ, май 2021.	1–5
1.4	Васильева М.А. ПРИМЕНЕНИЕ СИСТЕМЫ КОНТРОЛЯ ВЕРСИЙ GIT ДЛЯ ОРГАНИЗАЦИИ УЧЕБНОГО ПРОЦЕССА В ВУЗЕ // СОВРЕМЕННЫЕ ТЕХНОЛОГИИ В НАУКЕ И ОБРАЗОВАНИИ - СТНО-2021 Сборник трудов IV Международного научно-технического форума: в 10 т., Т. 10, 2021.	1-5

IV. Справочно-правовые системы.

Ссылки на Интернет-ресурсы:

- Pretty Git branch graphs [Электронный ресурс] // StackOverflow: [сайт]. [2021]. URL: <https://stackoverflow.com/a/9074343/17310482> (дата обращения: 13.03.2022)/;
- gitignore [Электронный ресурс] // git: [сайт]. URL: https://git-scm.com/docs/gitignore#_pattern_format (дата обращения: 14.03.2022).

Начальник Центра «Высшая школа
педагогического мастерства»

В. И. Модинец

Учебная программа подготовлена:

Доцентом кафедры
«Управление и защита информации»

М. А. Васильевой

« ____ » _____ 2022 г.