

**ЧАСТНОЕ ПРОФЕССИОНАЛЬНОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ
«НИЖЕГОРОДСКИЙ ГУМАНИТАРНО-ТЕХНИЧЕСКИЙ КОЛЛЕДЖ»
(ЧПОУ НГТК)**

РАССМОТРЕНО

на заседании Педагогического совета
Протокол № 9
от «05» мая 2026 г.

УТВЕРЖДАЮ



Директор ЧПОУ НГТК

Н.О. Ким

Приказ № 105/1 от «05» мая 2026 г.

РАБОЧАЯ ПРОГРАММА УЧЕБНОЙ ДИСЦИПЛИНЫ

**ОП.05 ИНФОРМАЦИОННЫЕ ТЕХНОЛОГИИ В ПРОФЕССИОНАЛЬНОЙ
ДЕЯТЕЛЬНОСТИ**

среднего профессионального образования
по программе подготовки специалистов среднего звена (ППССЗ)
09.02.12 «Техническая эксплуатация и сопровождение информационных систем»

Квалификация:
специалист по технической эксплуатации и
сопровождению информационных систем

Форма обучения: очная

Нормативный срок обучения:

1 год 10 месяцев на базе среднего общего образования

Рабочая программа дисциплины разработана на основе Федерального государственного образовательного стандарта среднего профессионального образования по специальности 09.02.12 «Техническая эксплуатация и сопровождение информационных систем».

Организация - разработчик: ЧПОУ НГТК

Разработчики: Чиненков А.Ф., преподаватель

СОДЕРЖАНИЕ

| | |
|---|-----------|
| 1. ОБЩАЯ ХАРАКТЕРИСТИКА РАБОЧЕЙ ПРОГРАММЫ УЧЕБНОЙ ДИСЦИПЛИНЫ | 4 |
| 2. СТРУКТУРА И СОДЕРЖАНИЕ УЧЕБНОЙ ДИСЦИПЛИНЫ | 5 |
| 3. УСЛОВИЯ РЕАЛИЗАЦИИ УЧЕБНОЙ ДИСЦИПЛИНЫ | 10 |
| 4. КОНТРОЛЬ И ОЦЕНКА РЕЗУЛЬТАТОВ ОСВОЕНИЯ УЧЕБНОЙ ДИСЦИПЛИНЫ | 11 |

1. ОБЩАЯ ХАРАКТЕРИСТИКА РАБОЧЕЙ ПРОГРАММЫ УЧЕБНОЙ ДИСЦИПЛИНЫ

1.1. Цель и место дисциплины в структуре основной образовательной программы:

Цель дисциплины «Информационные технологии в профессиональной деятельности»: например: формирование компетенций в области применения информационных технологий для решения профессиональных задач, освоение современных цифровых инструментов и развитие навыков их эффективного использования в профессиональной деятельности.

Дисциплина «Информационные технологии в профессиональной деятельности» включена в обязательную часть общепрофессионального цикла образовательной программы.

1.2. Планируемые результаты освоения дисциплины:

Результаты освоения дисциплины соотносятся с планируемыми результатами освоения образовательной программы, представленными в матрице компетенций выпускника (п. 4.3 ПОП).

В результате освоения дисциплины обучающийся должен:

| Код ОК | Умения | Знания | Владеть навыками |
|--------|--|---|------------------|
| ОК.02 | определять задачи для поиска информации, планировать процесс поиска, выбирать необходимые источники информации; применять средства информационных технологий для решения профессиональных задач; использовать современное программное обеспечение в профессиональной деятельности. | номенклатура информационных источников, применяемых в профессиональной деятельности; современные средства и устройства информатизации, порядок их применения; программное обеспечение в профессиональной деятельности, в том числе цифровые средства. | - |
| ОК.03 | определять траектории профессионального развития и самообразования. применять современную научную профессиональную терминологию. оценивать жизнеспособность проектной идеи. | возможные траектории профессионального развития и самообразования; основы предпринимательской деятельности, правовой и финансовой грамотности; основные этапы разработки и реализации проекта. | - |

2. СТРУКТУРА И СОДЕРЖАНИЕ УЧЕБНОЙ ДИСЦИПЛИНЫ

2.1. Объем учебной дисциплины и виды учебной работы

| Вид учебной работы | Объем в часах |
|---|----------------------|
| Объем образовательной программы учебной дисциплины | 42 |
| в т.ч. в форме практической подготовки | 22 |
| в т. ч.: | |
| теоретическое обучение | 14 |
| практические занятия | 22 |
| Промежуточная аттестация (экзамен) | 6 |

2.2. Тематический план и содержание учебной дисциплины

| Наименование разделов и тем | Примерное содержание учебного материала, практических и лабораторных занятий, курсовой проект (работа) | Объем, акад. ч. /в том числе в форме практической подготовки, акад. ч | Коды компетенций и личностных результатов, формированию которых способствует элемент программы |
|--|---|---|--|
| Тема 1.1. Искусственный интеллект как инструмент программиста | Содержание ИИ и LLM: зачем они нужны разработчику Обзор популярных ИИ-инструментов (GitHub Copilot, ChatGPT, Codeium) ИИ и написание кода: кейсы и ограничения Использование ИИ для генерации тестов, SQL-запросов Промпт-инжиниринг: формулировка запросов Ревью кода с ИИ: плюсы и минусы Генерация документации к проекту ИИ в CI/CD пайплайнах (оптимизация шагов) ChatOps: использование ботов в командной разработке Этические аспекты и ответственность при работе с ИИ | 7/4 3 | ОК 02, ОК 03 |
| | В том числе практических и лабораторных занятий | 4 | |
| | 1. Подключение и использование ChatGPT для генерации кода | | |
| | 2. Генерация автотестов на Python по описанию задачи | | |
| | 3. Написание SQL-запросов через Copilot | | |
| | 4. Рефакторинг кода с объяснением шагов | | |
| | 5. Генерация комментариев к функциям и классам | | |
| | 6. Сравнение работы нескольких ИИ-инструментов | | |
| | 7. Создание readme-файла проекта через ИИ | | |
| | 8. Написание GitHub Action с подсказками Copilot | | |
| | 9. Превращение баг-репорта в список задач | | |
| 10. Разработка промптов для сложных запросов | | | |
| Тема 1.2. Git и Markdown в командной разработке | Содержание | 7/4 | ОК 02, ОК 03 |
| | Контроль версий: зачем нужен Git Git: базовые команды, концепция веток Ветки, мерджи, pull request и конфликты GitHub/GitLab: интерфейс, CI, багтрекеры Markdown: синтаксис, структура, назначение | 3 | |

| | | | |
|--|--|------------|---------------------|
| | Документирование API в Markdown README.md как витрина проекта Использование GitHub Pages и Wiki Рецензирование кода через pull request Практика оформления задач и описаний | | |
| | В том числе практических и лабораторных занятий | 4 | |
| | 1. Создание и клонирование репозитория | | |
| | 2. Ведение истории коммитов и работа с ветками | | |
| | 3. Конфликт и его разрешение | | |
| | 4. Настройка CI в GitHub Actions | | |
| | 5. Создание красивого README.md | | |
| | 6. Использование маркдауна для changelog | | |
| | 7. Описание API-интерфейса в markdown | | |
| | 8. Работа с pull request и ревью кода | | |
| | 9. Создание и публикация проекта на GitHub Pages | | |
| | 10. Создание вики-проекта и структуры документации | | |
| Тема 1.3. Облачные сервисы и инструменты разработчика | Содержание | 7/4 | ОК 02, ОК 03 |
| | Основы работы с облаками: IaaS, PaaS, SaaS Яндекс Облако / VK Cloud / Selectel: обзор и интерфейс Хранилище, вычисления, базы данных в облаке Развёртывание приложения на облачном сервере Terraform / IaC: автоматизация инфраструктуры GitLab CI/CD + облако Облачные IDE (Replit, GitHub Codespaces) S3-хранилище и автоматизация бэкапов Логирование и мониторинг в облаке Безопасность облачных сред | 3 | |
| | В том числе практических и лабораторных занятий | 4 | |
| | 1. Регистрация и запуск виртуальной машины в Яндекс.Облаке | | |
| | 2. Развёртывание Python-приложения на облачном сервере | | |
| | 3. Использование S3-хранилища для логов | | |
| | 4. Настройка CI/CD-пайплайна для загрузки файлов | | |
| | 5. Подключение к облачной базе данных | | |
| | 6. Использование облачной IDE для командного проекта | | |

| | | | |
|--|--|------------|---------------------|
| | 7. Создание YAML-манифеста Terraform | | |
| | 8. Настройка доступа к bucket'у | | |
| | 9. Интеграция с логами и алертами | | |
| | 10. Аудит безопасности облачного проекта | | |
| Тема 1.4. Цифровые инструменты и экосистема разработчика | Содержание | 8/5 | ОК 02, ОК 03 |
| | IDE, расширения, сборщики: VS Code, JetBrains Bash и командная строка как инструмент Утилиты curl, wget, ping, telnet Форматы данных: JSON, YAML, XML Конфигурационные файлы и шаблоны DevTools в браузере и веб-отладка Task-менеджеры и трекеры: Trello, YouTrack Работа с docker-образами Инструменты тестирования API: Postman Автоматизация повседневных задач | 3 | |
| | В том числе практических и лабораторных занятий | 5 | |
| | 1. Работа в VS Code: настройка расширений | | |
| | 2. Написание bash-скрипта для автоматизации | | |
| | 3. Отправка API-запроса через curl и Postman | | |
| | 4. Разбор JSON-структуры и валидация | | |
| | 5. Написание dockerfile и сборка образа | | |
| | 6. Использование DevTools для анализа сайта | | |
| | 7. Создание задачи и доски в Trello | | |
| | 8. Отладка API на реальном сервисе | | |
| | 9. Настройка git hooks и lint-автоматизации | | |
| | 10. Создание шаблона конфига в YAML | | |
| Тема 1.5. Кибербезопасность и цифровая гигиена ИТ-специалиста | Содержание | 7/5 | ОК 02, ОК 03 |
| | Угрозы в разработке: инъекции, XSS, MITM Безопасные пароли, ключи, доступы Работа с .env-файлами и секретами Проверка зависимостей: Snyk, Dependabot Шифрование, хеширование и токены VPN, SSH и туннелирование Анонимизация и защита данных | 2 | |

| | | | |
|--|---|-----------|--|
| | Правила цифровой гигиены и GDPR Атаки на open-source проекты Повседневная безопасность в DevOps | | |
| | В том числе практических и лабораторных занятий | 5 | |
| | 1. Настройка SSH-ключей и безопасного подключения | | |
| | 2. Работа с .env-файлом в проекте | | |
| | 3. Сканирование зависимостей с Snyk | | |
| | 4. Пример XSS-атаки и защита от неё | | |
| | 5. Хеширование строки и проверка целостности | | |
| | 6. Шифрование данных с помощью openssl | | |
| | 7. Работа с GitHub Secrets и CI | | |
| | 8. Создание VPN-соединения | | |
| | 9. Формирование чек-листа цифровой гигиены | | |
| | 10. Анализ утечек и проверка паролей | | |
| | Промежуточная аттестация (экзамен) | 6 | |
| | Всего | 42 | |

3. УСЛОВИЯ РЕАЛИЗАЦИИ УЧЕБНОЙ ДИСЦИПЛИНЫ

3.1 Материально-техническое обеспечение

Реализация программы учебной дисциплины производится с применением дистанционных технологий и требует наличия электронной образовательной среды; учебного кабинета.

Оборудование учебного кабинета:

- классная доска;
- посадочные места по количеству обучающихся;
- рабочее место преподавателя;
- учебно-практическое оборудование, необходимое для проведения предусмотренных программой практических занятий. В соответствии с п.4.4 ФГОС СПО допускается замена оборудования его виртуальными аналогами.

Технические средства обучения:

- компьютеры с выходом в сеть Internet;
- сайт «Личная студия» с возможностью работы с электронным образовательным ресурсом;
- электронные библиотечные ресурсы.

Учебно-методическое обеспечение дисциплины:

- методические указания по организации практических занятий;
- методические указания по самостоятельной работе.

Программное обеспечение:

Программное обеспечение, являющееся частью электронной информационно-образовательной среды и базирующееся на телекоммуникационных технологиях:

- компьютерные обучающие программы;
- тренинговые и тестирующие программы;
- интеллектуальные роботизированные системы оценки качества выполненных работ;
- справочно-правовая система «Консультант плюс», «Гарант»;
- электронно-библиотечная система (ЭБС) ЭБС «IPR SMART» <http://iprbookshop.ru/>;
- программа управления образовательным процессом в ЭИОС (Информационная технология. Программа управления образовательным процессом. КОМБАТ).

3.2. Информационное обеспечение реализации программы

3.2.1. Основные печатные и/или электронные издания

1. .

4. КОНТРОЛЬ И ОЦЕНКА РЕЗУЛЬТАТОВ ОСВОЕНИЯ УЧЕБНОЙ ДИСЦИПЛИНЫ

| Результаты обучения | Критерии оценки | Методы оценки |
|--|---|--|
| <p>Знает:</p> <ul style="list-style-type: none">– основные понятия, функции, состав и принципы работы операционных систем;– архитектуры современных операционных систем;– особенности построения и функционирования семейств операционных систем "Unix" и "Windows";– принципы управления ресурсами в операционной системе;– основные задачи администрирования и способы их выполнения в изучаемых операционных системах. <p>Умеет:</p> <ul style="list-style-type: none">– управлять параметрами загрузки операционной системы;– выполнять конфигурирование аппаратных устройств;– управлять учетными записями, настраивать параметры рабочей среды пользователей;– управлять дисками и файловыми системами, настраивать сетевые параметры, управлять разделением ресурсов в локальной сети. | <ul style="list-style-type: none">– способность применять теоретические знания на практике при работе с различными операционными системами;– умение анализировать и решать задачи системного администрирования;– готовность к освоению новых технологий в области операционных систем и сред. | <p>Штудирование Тестирование Выполнение практических заданий Контрольная работа Выполнение заданий на экзамене</p> |