

Приложение ПМ.УП.01
к ОПОП по специальности
09.02.13 Интеграция
решений с применением
технологий
искусственного интеллекта
(квалификация –
специалист
по работе с искусственным
интеллектом)

РАБОЧАЯ ПРОГРАММА УЧЕБНОЙ ПРАКТИКИ
ПМ.01 РАЗРАБОТКА КОДА ДЛЯ ОБУЧЕНИЯ ИСКУССТВЕННОГО
ИНТЕЛЛЕКТА»

СОДЕРЖАНИЕ

ПМ.01 РАЗРАБОТКА КОДА ДЛЯ ОБУЧЕНИЯ ИСКУССТВЕННОГО ИНТЕЛЛЕКТА»	1
1. ПАСПОРТ ПРОГРАММЫ УЧЕБНОЙ ПРАКТИКИ	3
2. РЕЗУЛЬТАТЫ УЧЕБНОЙ ПРАКТИКИ	7
3. СТРУКТУРА И СОДЕРЖАНИЕ УЧЕБНОЙ ПРАКТИКИ	8
4. УСЛОВИЯ ОРГАНИЗАЦИИ И ПРОВЕДЕНИЯ УЧЕБНОЙ ПРАКТИКИ	10
5. КОНТРОЛЬ И ОЦЕНКА РЕЗУЛЬТАТОВ УЧЕБНОЙ ПРАКТИКИ	11

1. ПАСПОРТ ПРОГРАММЫ УЧЕБНОЙ ПРАКТИКИ

1.1 Область применения программы

Практическая подготовка - форма организации образовательной деятельности при освоении образовательной программы в условиях выполнения обучающимися определенных видов работ, связанных с будущей профессиональной деятельностью и направленных на формирование, закрепление, развитие практических навыков и компетенций по профилю соответствующей образовательной программы.

Образовательная деятельность в форме практической подготовки реализована как комплекс учебной и производственной практики в составе ОПОП СПО.

Программа учебной практики является частью ОПОП в соответствии с ФГОС СПО по специальности **09.02.13 Интеграция решений с применением технологий искусственного интеллекта** в части освоения основного вида профессиональной деятельности (ВПД) и соответствующих профессиональных компетенций (ПК):

Виды деятельности	Профессиональные компетенции, соответствующие видам деятельности
разработка кода для искусственного интеллекта	ПК 1.1. Формировать алгоритмы разработки программных модулей в соответствии с техническим заданием. ПК 1.2. Разрабатывать программные модули в соответствии с техническим заданием. ПК 1.3. Оформлять программный код в соответствии с техническим заданием. ПК 1.4. Использовать систему контроля версий программного кода с учетом обеспечения возможности организации групповой разработки. ПК 1.5. Выполнять отладку программных модулей с использованием специализированных программных средств. ПК 1.6. Выполнять тестирование программного кода. ПК 1.7. Составлять тестовые сценарии.

1.2 Цели и задачи учебной практики - требования к результатам освоения

С целью овладения указанным видом профессиональной деятельности и соответствующими профессиональными компетенциями обучающийся в ходе прохождения **учебной практики** должен:

Код ОК, ПК	Уметь	Знать	Владеть навыками
ПК 1.1	Анализировать	Основные методы и	Разработки,

	технические задания и выявлять требования к алгоритмам. Применять методы алгоритмизации для решения задач программирования. Разрабатывать оптимальные алгоритмы для решения задач в области ИИ.	подходы к построению алгоритмов (жадные алгоритмы, динамическое программирование, рекурсивные подходы). Принципы эффективной обработки данных. Языки программирования, применяемые для разработки алгоритмов (Python, C#, Java).	оптимизации и тестирования алгоритмов для ИИ-программ. Использования библиотек и инструментов для работы с алгоритмами и данными (Pandas, NumPy, Scikit-learn). Применения структур данных (деревья, графы, списки) для реализации алгоритмов.
ПК 1.2	Реализовывать программные модули на основе требований технического задания. Писать чистый, понятный и поддерживаемый код. Использовать стандартные библиотеки и фреймворки для ускорения разработки.	Принципы модульного программирования. Языки программирования для разработки модулей (Python, C#, Java). Стандартные фреймворки и библиотеки для работы с ИИ (TensorFlow, PyTorch, Keras).	Разработки модульных ИИ-систем, соответствующих требованиям производительности и безопасности. Внедрения разработанных ИИ-модулей в комплексные программные системы. Оптимизации кода и работы с интерфейсами для взаимодействия между модулями.
ПК 1.3	Оформлять код в соответствии с принятыми стандартами и требованиями. Документировать разработанный программный код. Применять соглашения о наименованиях переменных, функций и классов (например, PEP8 для Python).	Основные принципы чистого кода (Clean Code). Стандарты и практики документирования программного обеспечения. Инструменты для автоматической проверки качества кода (например, PyLint, ESLint).	Оформления, документирования и структурирования кода для последующей поддержки. Использования инструментов статического анализа кода для выявления ошибок и улучшения качества. Работы с системами документирования кода (например, Doxygen, Sphinx).
ПК 1.4	Работать с системами контроля версий для управления проектами (Git, GitLab). Организовывать совместную работу над проектом через ветки	Принципы работы распределенных систем контроля версий. Основные команды и операции в Git (commit, pull, push, merge).	Управления проектами с использованием Git для организации командной работы. Разрешения конфликтов при слиянии веток и

	разработки и слияние изменений. Разрешать конфликты при слиянии кода.	Методы разрешения конфликтов в ходе групповой разработки.	использования pull request для рецензирования кода. Настройки процессов CI/CD для автоматического тестирования и развертывания кода.
ПК 1.5	Использовать инструменты для отладки программного кода. Идентифицировать и исправлять ошибки в программе. Применять методы логирования для анализа выполнения программ.	Принципы работы отладчиков и логирования. Способы выявления ошибок в программе (отладка по шагам, точки останова). Инструменты для отладки кода (например, PyCharm, Visual Studio Debugger).	Отладки программных модулей с использованием пошаговой проверки. Применения методов логирования и профилирования производительности. Использования специальных средств для отладки многопоточных программ.
ПК 1.6	Проводить различные виды тестирования (юнит-тестирование, интеграционное тестирование). Разрабатывать тестовые сценарии для проверки корректности работы программных модулей. Автоматизировать тестирование программного обеспечения.	Принципы тестирования программного обеспечения. Методы и подходы к написанию тестов (Test-Driven Development, Behavior-Driven Development). Инструменты для тестирования программного кода (PyTest, JUnit, Selenium).	Написания юнит-тестов для проверок отдельных функций и модулей. Создания автоматизированных тестов для интеграционных проверок. Работы с CI/CD пайплайнами для автоматизации тестирования.
ПК 1.7	Определять критические сценарии работы системы, которые необходимо протестировать. Разрабатывать пошаговые тестовые сценарии на основе требований. Оценивать покрытие тестов и их соответствие техническому заданию.	Основы тест-дизайна и методы разработки тестовых сценариев. Принципы проектирования сценариев для функционального и нефункционального тестирования. Методы составления тест-кейсов для разных типов тестирования.	Проектирования тестовых сценариев, включая пограничные и негативные сценарии. Использования шаблонов для написания тест-кейсов. Автоматизации создания и выполнения тестовых сценариев.

1.3 Количество часов (недель) на учебную практику всего

– 36 час. (1 неделя)

2. РЕЗУЛЬТАТЫ УЧЕБНОЙ ПРАКТИКИ

Результатом **учебной практики** является овладение обучающимися видом профессиональной деятельности (ПК) компетенциями:

Виды деятельности	Профессиональные компетенции, соответствующие видам деятельности
разработка кода для искусственного интеллекта	ПК 1.1. Формировать алгоритмы разработки программных модулей в соответствии с техническим заданием. ПК 1.2. Разрабатывать программные модули в соответствии с техническим заданием. ПК 1.3. Оформлять программный код в соответствии с техническим заданием. ПК 1.4. Использовать систему контроля версий программного кода с учетом обеспечения возможности организации групповой разработки. ПК 1.5. Выполнять отладку программных модулей с использованием специализированных программных средств. ПК 1.6. Выполнять тестирование программного кода. ПК 1.7. Составлять тестовые сценарии.

3. СТРУКТУРА И СОДЕРЖАНИЕ УЧЕБНОЙ ПРАКТИКИ

3.1. Тематический план практики

Коды формируемых компетенций	Наименование ПМ	Объем времени, отводимый на практику (час./нед.)	Сроки проведения
ПК 1.1-1.7	разработка кода для искусственного интеллекта	36 час., 1 нед.	2 семестр

Наименование МДК	Наименования тем практики	Количество часов по темам
МДК 01.01 Разработка программных модулей в системах искусственного интеллекта	1. Сбор и предобработка данных из открытых источников для задач машинного обучения. 2. Разработка простых программных модулей для анализа данных с использованием библиотек Python (Pandas, NumPy). 3. Разработка базовых моделей машинного обучения (линейная регрессия, дерево решений) для реальных задач. 4. Визуализация данных и результатов работы моделей ИИ с использованием Matplotlib. 5. Интеграция предобученной модели машинного обучения в простое мобильное приложение (Android Studio). 6. Разработка прототипа мобильного приложения с элементами ИИ (например, распознавание объектов).	12
МДК 01.02 Разработка программных модулей в системах искусственного интеллекта		12
МДК 01.03		12

Тестирование программных модулей	7. Написание и отладка юнит-тестов для программных модулей, реализованных в ИИ-системах. 8. Работа с системами контроля версий (Git, GitHub) для управления проектами. 9. Контейнеризация простых ИИ-приложений с использованием Docker. 10. Внедрение и отладка CI/CD процессов для автоматизированного тестирования.	
----------------------------------	---	--

4. УСЛОВИЯ ОРГАНИЗАЦИИ И ПРОВЕДЕНИЯ УЧЕБНОЙ ПРАКТИКИ

4.1. Требования к материально-техническому обеспечению

Базы практики (мастерские), оснащенные:

Автоматизированные рабочие места по количеству обучающихся (процессор не ниже Core i3, оперативная память объемом не менее 4 Гб или аналоги)

Автоматизированное рабочее место преподавателя (процессор не ниже Core i3, оперативная память объемом не менее 4 Гб или аналоги)

Мультимедийный проектор

Мультимедийный экран

Средства аудиовизуализации

Комплект учебно-методических материалов

Пакет прикладных программ

СУБД

Точка доступа

Коммутатор

Маршрутизатор

Программное обеспечение для разработки:

– Python (для разработки программных модулей и работы с ИИ).

– TensorFlow, Keras, PyTorch (для работы с моделями машинного обучения и нейронными сетями).

– Jupyter Notebook (для интерактивной разработки и анализа данных).

– Git (для работы с системами контроля версий).

Программное обеспечение для моделирования и тестирования:

– UML инструменты (например, Enterprise Architect или Visual Paradigm).

– Эмуляторы для тестирования решений в области ИТ-инфраструктуры.

4.2. Информационное обеспечение реализации программы

1. Джонс, М. Т. Программирование искусственного интеллекта в приложениях / М. Т. Джонс ; перевод А. И. Осипов. — 2-е изд. — Саратов : Профобразование, 2024. — 312 с. — ISBN 978-5-4488-0116-7. — Текст : электронный // Электронный ресурс цифровой образовательной среды СПО PROобразование : [сайт]. — URL: <https://profspo.ru/books/89866>

2. Котляров, В. П. Основы тестирования программного обеспечения : учебное пособие для СПО / В. П. Котляров. — Саратов : Профобразование, 2024 — 335 с. — ISBN 978-5-4488-0364-2. — Текст : электронный // Электронный ресурс цифровой образовательной среды СПО PROобразование : [сайт]. — URL: <https://profspo.ru/books/86202>

3. Федорова Г.Н. Разработка модулей программного обеспечения для компьютерных систем: учебник. – М.: Издательский центр «Академия», 2020. – 384 с.

4.3 Общие требования к организации учебной практики

Обязательным условием направления на учебную практику в рамках профессионального модуля **ПМ. 01 разработка кода для искусственного интеллекта** является освоение в полном объеме МДК, входящим в его состав

4.4. Кадровое обеспечение образовательного процесса

4.4.1. Реализация образовательной программы обеспечивается педагогическими работниками колледжа, а также лицами, привлекаемыми к реализации образовательной

программы на условиях гражданско-правового договора, в том числе из числа руководителей и работников организаций, направление деятельности которых соответствует области профессиональной деятельности [06](#) Связь, информационные и коммуникационные технологии (имеющих стаж работы в данной профессиональной области не менее 3 лет).

4.4.2. Квалификация педагогических работников колледжа отвечает квалификационным требованиям, указанным в квалификационных справочниках, и (или) профессиональных стандартах (при наличии).

Педагогические работники, привлекаемые к реализации образовательной программы, получают дополнительное профессиональное образование по программам повышения квалификации, в том числе в форме стажировки в организациях направление деятельности которых соответствует области профессиональной деятельности - 06 Связь, информационные и коммуникационные технологии, не реже 1 раза в 3 года с учетом расширения спектра профессиональных компетенций.

Доля педагогических работников (в приведенных к целочисленным значениям ставок), обеспечивающих освоение обучающимися профессиональных модулей, имеющих опыт деятельности не менее 3 лет в организациях, направление деятельности которых соответствует области профессиональной деятельности - 06 Связь, информационные и коммуникационные технологии, в общем числе педагогических работников, реализующих образовательную программу, должна быть не менее 25 процентов.

5. КОНТРОЛЬ И ОЦЕНКА РЕЗУЛЬТАТОВ УЧЕБНОЙ ПРАКТИКИ

Формой отчетности студента по учебной практике является письменный отчет о выполнении работ и приложений к отчету, свидетельствующих о закреплении знаний, умений, приобретении практического опыта, формировании профессиональных компетенций, освоении профессионального модуля.

По результатам защиты студентами отчетов выставляется дифференцированный зачет по учебной практике.

Код ПК, ОК	Критерии оценки результата (показатели освоенности компетенций)	Формы контроля и методы оценки
ПК.1.1 Формировать алгоритмы разработки программных модулей в соответствии с техническим заданием.	Оценка "отлично": алгоритмы разработаны в полном соответствии с ТЗ, оптимизированы и понятны. Оценка "хорошо": алгоритмы разработаны в соответствии с ТЗ, но допускают незначительные отклонения. Оценка "удовлетворительно": алгоритмы разработаны с частичным соответствием ТЗ.	Защита отчёта с демонстрацией разработанных алгоритмов и устное собеседование.
ПК.1.2 Разрабатывать программные модули в соответствии с техническим заданием.	Оценка "отлично": программные модули разработаны в полном соответствии с ТЗ, тесты проходят успешно. Оценка "хорошо": программные модули разработаны с минимальными несоответствиями, тесты в целом успешны. Оценка "удовлетворительно": программные модули разработаны с существенными доработками.	Защита отчёта по разработанным модулям, проверка выполнения ТЗ и прохождения тестов.

ПК.1.3 Оформлять программный код в соответствии с техническим заданием.	Оценка "отлично": код полностью оформлен в соответствии с требованиями, включая комментарии и стиль кода. Оценка "хорошо": код оформлен в соответствии с требованиями, допускаются мелкие недочёты. Оценка "удовлетворительно": код оформлен частично в соответствии с требованиями.	Проверка оформленного кода, соблюдения стиля и соответствия установленным требованиям.
ПК.1.4 Использовать систему контроля версий программного кода с учетом обеспечения возможности организации групповой разработки.	Оценка "отлично": система контроля версий используется эффективно, изменения фиксируются корректно. Оценка "хорошо": система контроля версий используется, но имеются мелкие нарушения порядка фиксации изменений. Оценка "удовлетворительно": система контроля версий используется частично или с ошибками.	Проверка использования системы контроля версий (репозиторий), демонстрация фиксации изменений.
ПК.1.5 Выполнять отладку программных модулей с использованием специализированных программных средств.	Оценка "отлично": отладка выполнена полностью, ошибки устранены, работа модулей оптимизирована. Оценка "хорошо": отладка выполнена, ошибки устранены, но оптимизация частичная. Оценка "удовлетворительно": отладка выполнена частично, ошибки устранены не полностью.	Демонстрация процесса отладки с использованием инструментов, отчёт по устранённым ошибкам.
ПК.1.6 Выполнять тестирование программного кода.	Оценка "отлично": тестирование выполнено в полном объёме, тесты соответствуют ТЗ, выявленные ошибки исправлены. Оценка "хорошо": тестирование выполнено, тесты соответствуют ТЗ, незначительные ошибки остались. Оценка "удовлетворительно": тестирование выполнено частично, ошибки выявлены, но не исправлены.	Предоставление отчёта о тестировании, демонстрация успешного прохождения тестов.
ПК.1.7 Составлять тестовые сценарии.	Оценка "отлично": тестовые сценарии составлены полностью, покрывают все функциональные требования. Оценка "хорошо": тестовые сценарии составлены, но не покрывают незначительную часть функциональных требований. Оценка "удовлетворительно": тестовые сценарии составлены частично, покрывают минимальный функционал.	Предоставление тестовых сценариев, проверка их соответствия функциональным требованиям.