Федеральное государственное бюджетное образовательное учреждение
высшего образования
«Новосибирский государственный технический университет»
Кафедра автоматизированных систем управления
Кафедра автоматики
Кафедра вычислительной техники
Кафедра защиты информации

Паспорт
расчетно-графической работы

по дисциплине «Технологии и методы программирования», 4 семестр

Методика оценки

Выполнение расчетно-графической работы (далее - РГР) является формой текущей аттестации (контроля) по дисциплине, предусмотренной учебным планом.

Цель РГР: освоение технологий разработки программного приложения, включая интерфейс, базу данных, архитектуру, API, контроль версий и тестирование; обоснованный выбор программных средств и применение методов разработки с соблюдением норм информационной безопасности по индикаторам ОПК-2.1 и ОПК-2.2.

Обязательным элементом РГР является решение профессиональных задач разработки и проверки программ в соответствии с выбранной темой.

Номер задания из перечня п. 4 определяется преподавателем. Для каждого студента задаются индивидуальные исходные данные и параметры; тема и исходные данные фиксируются в задании на РГР.

РГР выполняется индивидуально.

Количество тем и индивидуальных наборов исходных данных обеспечивает каждого обучающегося индивидуальным заданием РГР.

Замена задания РГР осуществляется по согласованию с преподавателем из числа резервных (не занятых) заданий.

Преподаватель осуществляет руководство по выполнению задания, оказывает консультационную помощь и принимает отчет по РГР.

По результатам выполнения РГР выполняется отчет, который состоит из следующих частей:

Титульный лист.

Введение: цель, постановка задания, исходные данные и используемые источники.

Теоретическая часть: описание выбранных методов, алгоритмов и программных средств.

Практическая часть: схема данных, архитектура, интерфейс и реализация, история изменений, тестовые сценарии и результаты; для нейросетевой темы — данные, метрики и ограничения модели.

Выводы или заключение.

Список литературы.

Приложения (при необходимости).

Требования к оформлению:

Объем РГР до 50 страниц машинописного текста формата А4. Шрифт Times New Roman, 12. Формулы набираются в редакторе Math Type. Размещение сканированных формул не допускается. Нумерация страниц сквозная, в нижней части листа по центру арабскими цифрами. Работа должна быть отредактирована, не содержать орфографических, синтаксических и стилевых ошибок.

Отчет в установленные сроки сдается на кафедру для проверки. Преподаватель оценивает качество работы, отмечает положительные стороны и недостатки работы и определяет, допускается ли она к защите. При необходимости преподаватель возвращает РГР студенту для доработки и устанавливает сроки повторного предоставления для проверки. До защиты работы студентом должны быть сделаны необходимые исправления и дополнения по всем замечаниям преподавателя.

При положительном результате оценивания РГР студент её распечатывает, передает на кафедру и защищает до сессии в назначенное преподавателем время.

Защита РГР состоит в индивидуальном устном собеседовании студента с преподавателем. В процессе защиты выявляется уровень знаний студента, степень его самостоятельности при выполнении работы. По результатам защиты студенту выставляется оценка в соответствии с критериями, приведенными в п. 2 настоящего Паспорта.

Критерии оценки результатов обучения, соотнесенных с уровнями освоения индикаторов достижения компетенций

Общие правила выставления оценок текущей аттестации определяются балльно-рейтинговой системой, установленной локальным актом НГТУ.

РГР считается выполненной **на продвинутом** уровне, если все необходимые алгоритмы реализованы самостоятельно и без ошибок; все разделы РГР выполнены правильно и в полном объеме; работа оформлена в соответствии с требованиями; сдана преподавателю в указанные сроки и не возвращалась для доработки; даны полные и развернутые выводы и рекомендации; на защите студентом даны уверенные и аргументированные ответы. Совокупность запланированных результатов и соотнесенных с ними индикаторов достижения компетенций не содержит пробелов. Закрепленные за РГР компетенции сформированы на продвинутом уровне. Оценка составляет *от 18 до 20 баллов.*

РГР считается выполненной **на базовом** уровне, если все необходимые алгоритмы реализованы самостоятельно и без существенных ошибок; все разделы РГР выполнены правильно, но есть замечания к полноте предоставления информации; работа оформлена в соответствии с требованиями; сдана преподавателю в указанные сроки и однократно возвращалась студенту для незначительной доработки; в заключении даны выводы и рекомендации; на защите студентом допущены непринципиальные ошибки. Совокупность запланированных результатов и соотнесенных с ними индикаторов достижения компетенций содержит несущественные пробелы. Закрепленные за РГР компетенции сформированы на базовом уровне. Оценка составляет *от 14 до 17 баллов.*

РГР считается выполненной **на пороговом** уровне, если все необходимые алгоритмы реализованы самостоятельно, но с ошибками, часть из которых носит принципиальный характер; есть замечания к полноте предоставления информации; работа оформлена в соответствии с требованиями; сдана преподавателю, но неоднократно возвращалась студенту для доработки; в заключении даны краткие выводы; защита РГР вызывает у студента серьезные затруднения. Совокупность запланированных результатов и соотнесенных с ними индикаторов достижения компетенций содержит существенные пробелы. Закрепленные за РГР компетенции сформированы на пороговом уровне. Оценка составляет *от 10 до 13 баллов.*

РГР считается **не выполненной** (ниже порогового уровня), если алгоритмы реализованы с серьезными ошибками; есть замечания к полноте предоставления информации и оформлению; РГР была сдана преподавателю, но неоднократно возвращалась студенту для доработки, что не привело к улучшению ее качества; РГР не допущена до защиты, что свидетельствует о неудовлетворительном уровне достигнутых студентом результатов. Совокупность запланированных результатов и соотнесенных с ними индикаторов достижения компетенций содержит множественные существенные пробелы. Закрепленные за РГР компетенции не сформированы. Оценка составляет менее 10 баллов.

Шкала оценки

В общей оценке по дисциплине баллы за РГР учитываются в соответствии с правилами балльно-рейтинговой системы, приведенными в рабочей программе дисциплины.

РГР как форма текущей аттестации (контроля) по дисциплине считается успешно выполненной, если сумма полученных баллов составляет от 10 до 20 баллов включительно.

Примерный перечень тем РГР

1. Веб-каталог учебных материалов. Спроектировать реляционную БД, реализовать адаптивный интерфейс, поиск и API; проверить ограничения данных и негативные сценарии.
2. Система учёта заявок на обслуживание. Разработать модель состояний заявки, слоистую архитектуру и REST API; проверить допустимые переходы и целостность БД.
3. Сервис бронирования оборудования. Спроектировать связанные таблицы, интерфейс выбора времени и транзакционную проверку конфликтов; протестировать одновременные запросы.
4. Электронный журнал лабораторных работ. Реализовать модель данных, интерфейс ввода и отчётные SQL-запросы; проверить диапазоны значений и согласованность записей на синтетических данных.
5. Планировщик задач. Разработать компонентный интерфейс с фильтрацией, backend и хранение задач; сравнить подходы Scrum и Kanban для организации собственной работы.
6. Каталог программных компонентов. Реализовать учёт версий и зависимостей, API и поиск; обосновать выбор структур данных, архитектуры и инструментов разработки.
7. Информационная система небольшой библиотеки. Спроектировать нормализованную БД, выдачу и возврат книг, интерфейс и тесты целостности; использовать вымышленные записи читателей.

8. Система учёта складских остатков. Реализовать операции поступления и списания в транзакциях, SQL-отчёты и веб-интерфейс; протестировать запрет отрицательных остатков.
9. Сервис проведения опросов. Разработать хранение вопросов и синтетических ответов, API и диаграмму итогов; проверить валидацию и корректность агрегатных запросов.
10. Панель анализа синтетических журналов событий. Реализовать загрузку данных, фильтры, группировки и индексы; измерить выполнение запросов и проверить безопасное отображение содержимого.
11. Приложение сравнения методов поиска. Реализовать интерфейс эксперимента, backend и хранение результатов; обосновать типизацию, структуры данных и тесты граничных случаев.
12. API справочника организаций. Спроектировать контракт, пагинацию, поиск и обработку ошибок; документировать API и автоматизировать интеграционные тесты с тестовой БД.
13. Рефакторинг учебного приложения с паттернами. Выделить слои и применить обоснованно выбранные паттерны; зафиксировать изменения в системе контроля версий и подтвердить поведение регрессионными тестами.
14. Сравнение frontend-подходов. Реализовать одинаковый небольшой интерфейс на чистом JavaScript и с выбранным фреймворком; сравнить компоненты, состояние и автоматизацию тестов.
15. Проектирование и оптимизация базы данных каталога. Построить схему и SQL-запросы, сравнить планы до и после индексации; представить результат через простой API и интерфейс.
16. Автоматизация тестирования веб-приложения. Для согласованного учебного приложения разработать модульные, интеграционные и системные проверки, набор граничных случаев и отчёт о дефектах.
17. Нейросетевой классификатор синтетических сообщений. Подготовить данные, обучающую и проверочную выборки, оценить метрики; подключить модель через API и показать результат в интерфейсе.
18. Прогнозирование синтетической временной последовательности. Разработать нейросетевой прототип, сравнить с простым базовым методом, организовать хранение данных и визуализацию результатов.
19. Сервис рекомендаций по учебному каталогу. Создать небольшой нейросетевой прототип на обезличенных синтетических данных, API и интерфейс; оценить качество, ограничения и обработку ошибок.
20. Безопасный учебный сервис регистрации обращений. Реализовать валидацию, параметризованные SQL-запросы и разграничение тестовых ролей; проверить API и интерфейс автоматизированными негативными тестами.