

Министерство науки и высшего образования Российской Федерации
НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ
ТОМСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ (НИ ТГУ)

Институт прикладной математики и компьютерных наук

УТВЕРЖДЕНО:

Директор

А. В. Замятин



Оценочные материалы по дисциплине

Введение в программную инженерию

по направлению подготовки

02.04.02 Фундаментальная информатика и информационные технологии

Направленность (профиль) подготовки:
Системная инженерия и управление IT-проектами

Форма обучения

Очная

Квалификация

Магистр

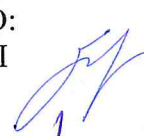
Год приема

2025

СОГЛАСОВАНО:

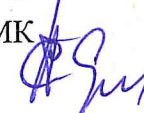
Руководитель ОП

А.Н. Моисеев



Председатель УМК

С.П. Сущенко



Томск – 2025

1. Компетенции и индикаторы их достижения, проверяемые данными оценочными материалами

Целью освоения дисциплины является формирование следующих компетенций:

ОПК-2 Способен применять компьютерные/суперкомпьютерные методы, современное программное обеспечение (в том числе отечественного производства) для решения задач профессиональной деятельности.

ОПК-5 Способен устанавливать и сопровождать программное обеспечение информационных систем, осуществлять эффективное управление разработкой программных средств и проектов.

УК-2 Способен управлять проектом на всех этапах его жизненного цикла.

Результатами освоения дисциплины являются следующие индикаторы достижения компетенций:

ИОПК-2.1 Обладает необходимыми знаниями основных концепций современных вычислительных систем и программного обеспечения (в том числе отечественного производства)

ИОПК-2.2 Использует методы высокопроизводительных вычислительных технологий, современного программного обеспечения (в том числе отечественного производства) для решения задач профессиональной деятельности

ИОПК-2.3 Использует инструментальные средства высокопроизводительных вычислений в научной и практической деятельности

ИОПК-5.3 Осуществлять эффективное управление разработкой программных средств и проектов

ИУК-2.1 Формулирует цель проекта, обосновывает его значимость и реализуемость.

ИУК-2.2 Разрабатывает программу действий по решению задач проекта с учетом имеющихся ресурсов и ограничений.

ИУК-2.3 Обеспечивает выполнение проекта в соответствии с установленными целями, сроками и затратами.

2. Оценочные материалы текущего контроля и критерии оценивания

Элементы текущего контроля:

- практическая работа (проект);
- зачет.

Задания для практической работы (ИОПК-2.1, ИОПК-2.2, ИОПК-2.3, ИОПК-5.3, ИУК-2.1, ИУК-2.2, ИУК-2.3)

Можно выбрать проект из предложенных категорий:

1. Клиент для автоматизированного доступа к метапоисковой системе Searx.

Необходимо производить поисковый запрос к любой копии системы с передачей параметров и парсить результат. Язык реализации Java.

2. Клиент для любого облачного хранилища файлов (Яндекс.Диск, Google Drive и пр.).

3. Взять любой API с сайта Rapid API и сделать для него клиент. Для бакалавров предпочтительным способом является библиотека Java, построенная по разумным архитектурным принципам. Для магистрантов допустимо минимальное консольное приложение на Python или Node.js.

Критерии оценивания:

Зачёт за практическую работу ставится, если программа функционирует в соответствии с требованиями, и автор способен пояснить написанный код.

3. Оценочные материалы итогового контроля и критерии оценивания

Экзаменационный билет состоит из трех устных вопросов, необходимо дать на них развернутый ответ.

Перечень тем для теоретических вопросов (ИОПК-2.1):

1. Цель и задачи программной инженерии.
 2. Понятия проекта, реализации проекта, методологии и метода.
 3. Историческая классификация методологий разработки ПО.
 4. История Унифицированного процесса.
 5. Понятия варианта использования.
 5. Понятие архитектуры.
 6. Итеративность и инкрементность.
 7. Фазы Унифицированного процесса.
 8. Определение и назначение вариантов использования.
 9. Понятия актантов и ролей.
 10. Классы и их типы.
 11. Персонал, проект, продукт, процесс.
 12. Оценка и классификация рисков.
 13. Архитектурные блоки и образцы.
 14. Порядок разработки архитектуры.
 15. Клиент-серверная архитектура.
 16. Микросервисная архитектура.
 17. Асинхронный ввод/вывод.
 18. Распределённые системы.
 19. Одноранговая архитектура.
 20. Распределённые хеш-таблицы.
 21. Модель реализации, билды и итерации.
 22. Модель тестирования и тестовые примеры.
 23. Перспективные практики тестирования.
 24. Behaviour-Driven Development и Behave.
 25. PlantUML .
 26. История гибких методологий разработки.
 27. Ценности и принципы Agile.
 28. Scrum.
 29. Идея UML, достоинства и недостатки.
 30. Диаграммы UML: вариантов использования, компонентов, классов, последовательности и активности.
 31. Другие типы диаграмм: Ганга, IDEF0 и DFD.
 32. Extreme Programming.
 33. Rapid Application Development.
 34. Feature Driven Development.
 35. Dynamic Systems Development Method.
 36. История зарождения технологического предпринимательства.
 37. Правовой статус и жизненный цикл стартапа.
 38. Типы финансирования и логика инвестора.
 39. Особенности развития научных проектов и проектов СПО.
- Критерии оценивания:

Зачет выставляется, если сдан проект, на все теоретические вопросы даны правильные развернутые ответы.

4. Оценочные материалы для проверки остаточных знаний (сформированности компетенций)

Теоретические вопросы (ИОПК-2.1)

1. Цель и задачи программной инженерии.
2. Понятие архитектуры.
3. Итеративность и инкрементность.
4. Модель тестирования и тестовые примеры.
5. Behaviour-Driven Development и Behave.
6. Диаграммы UML: вариантов использования, компонентов, классов, последовательности и активности.
7. Другие типы диаграмм: Ганга, IDEF0 и DFD.
8. Feature Driven Development.
9. Dynamic Systems Development Method.

Информация о разработчиках

Пожидаев Михаил Сергеевич, канд. техн. наук, доцент кафедры теоретических основ информатики.