

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

Архитектура программных систем

по направлению подготовки

01.04.02 Прикладная математика и информатика

Направленность (профиль) подготовки:
Информационная безопасность

Форма обучения
Очная

Квалификация
Магистр

Год приема
2025

СОГЛАСОВАНО:
Руководитель ОП
А.Ю. Матророва

Председатель УМК
С.П. Сущенко

Томск – 2025

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

Целью освоения дисциплины является формирование следующих компетенций:

ПК-1 Способен формализовать требования к программному обеспечению, спроектировать программное обеспечение, написать программный код, а также проверить работоспособность программного обеспечения и исправить дефекты.

Результатами освоения дисциплины являются следующие индикаторы достижения компетенций:

ИПК-1.1 Осуществляет анализ требований к программному обеспечению, построение формальной модели, проверку работоспособности программного обеспечения и исправление дефектов.

ИПК-1.2 Осуществляет разработку технических спецификаций на программные компоненты и их взаимодействие, разработку процедур проверки работоспособности и измерения характеристик программного обеспечения, разработку тестовых наборов данных.

ИПК-1.3 Осуществляет проектирование программного обеспечения, работу с системой контроля версий, рефакторинг и оптимизацию программного кода.

2. Оценочные материалы текущего контроля и критерии оценивания

Элементы текущего контроля:

- лабораторная работа (проект);
- зачет.

Задания для лабораторной работы (ИПК-1.1, ИПК-1.2)

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

3. Оценочные материалы итогового контроля (промежуточной аттестации) и критерии оценивания

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

Перечень тем для теоретических вопросов (ИПК-1.1, ИПК-1.2, ИПК-1.3):

1. История развития архитектур.
2. Ранний этап и мейнфреймы.
3. Эпоха клиент-серверных систем.
4. Переход к сервисной и микросервисной архитектурам.

5. Роль одноранговой архитектуры.
6. Распределённые и одноранговые системы.
7. Общая характеристика распределённых систем, их сильные и слабые стороны.
8. Apache Hadoop и Apache Spark.
9. Общая характеристика одноранговых систем.
10. Bitcoin, DHT и другие примеры.
11. Теорема CAP.
12. Сервис-ориентированная архитектура.
13. Общая характеристика сервис-ориентированной архитектуры.
14. Микросервисная архитектура.
15. Технологии микросервисной архитектуры.
16. Брокеры сообщений для микросервисной архитектуры.
17. Архитектуры операционных систем.
18. Требования к архитектуре и компоненты операционных систем.
19. Монолитная, микроядерная, гибридная и эгзоядерная архитектуры.
20. Примеры операционных систем для различных архитектур.

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 (ИПК-1.1, ИПК-1.2, ИПК-1.3):

1. История развития архитектур.
2. Ранний этап и мейнфреймы.
3. Эпоха клиент-серверных систем.
4. Переход к сервисной и микросервисной архитектурам.
5. Общая характеристика распределённых систем, их сильные и слабые стороны.
6. Apache Hadoop и Apache Spark.
7. Bitcoin, DHT и другие примеры.
8. Сервис-ориентированная архитектура.
9. Общая характеристика сервис-ориентированной архитектуры.
10. Микросервисная архитектура.
11. Требования к архитектуре и компоненты операционных систем.

Информация о разработчиках

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