



**МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ
РОССИЙСКОЙ ФЕДЕРАЦИИ**

федеральное государственное бюджетное образовательное учреждение
высшего образования
**«ИРКУТСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ»
(ФГБОУ ВО «ИГУ»)**

Институт математики и информационных технологий
Кафедра информационных технологий



Рабочая программа дисциплины (модуля)

**Б1.О.1.07 Современные технологии разработки программного
обеспечения**

Направление подготовки	09.04.03 Прикладная информатика
Направленность (профиль) подготовки	Комплексные информационные системы
Квалификация выпускника	магистр
Форма обучения	очная

Иркутск 2024 г.

1. ЦЕЛИ И ЗАДАЧИ ДИСЦИПЛИНЫ

Цель

Формирование набора компетенций необходимых для разработки сложноструктурированных многокомпонентных децентрализованных распределенных информационных и экономических систем и приложений

Задачи:

- Формирование понимания экономических и технологических аспектов разработки децентрализованных распределенных систем;
- Формирование навыков организации рабочего окружения;
- Формирования навыков разработки, отладки, тестирования, развертывания и сопровождения децентрализованных распределенных систем
- Формирование навыков командной работы: декомпозиция задач по областям ответственности, кооперация, персональная ответственность

2. МЕСТО ДИСЦИПЛИНЫ В СТРУКТУРЕ ОПОП ВО

2.1. Учебная дисциплина (модуль) относится к обязательной части программы и изучается на первом курсе.

2.2. Для изучения данной учебной дисциплины (модуля) необходимы знания, умения и навыки, сформированные следующими дисциплинами:

- Проектирование информационных систем;
- Разработка веб-приложений;
- Информационные системы и технологии;
- Языки программирования;
- Структуры данных.

2.3. Перечень последующих учебных дисциплин, для которых необходимы знания, умения и навыки, формируемые данной учебной дисциплиной:

- Технологическая (проектно-технологическая) практика
- Преддипломная практика

3. ТРЕБОВАНИЯ К РЕЗУЛЬТАТАМ ОСВОЕНИЯ ДИСЦИПЛИНЫ

Процесс освоения дисциплины направлен на формирование компетенций (элементов следующих компетенций) в соответствии с ФГОС ВО по соответствующему направлению подготовки.

Перечень планируемых результатов обучения по дисциплине (модулю), соотнесенных с индикаторами достижения компетенций

Компетенция	Индикаторы компетенций	Результаты обучения
ПК-2 Способность разрабатывать и адаптировать прикладное программное обеспечение.	ИДК ПК2.1 Способен записывать программный код в соответствии с требуемой парадигмой программирования	Умеет записывать программный код в соответствии с требуемой парадигмой программирования.
	ИДК ПК2.2 Способен выбирать и анализировать способы	Умеет выполнять подбор и анализ способов для решения прикладных задач.

	решения прикладных задач,	
	ИДК ПК2.3 Способен адаптировать и модифицировать программное обеспечение в соответствии с требованиями	Умеет адаптировать и модифицировать программное обеспечение в соответствии с требованиями
ПК-4 Готовность к включению профессиональное сообщество; способность проводить под научным руководством локальные исследования на основе существующих методов в конкретной области профессиональной деятельности	ИДК ПК4.1 Имеет навык подготовки и проведения публичных докладов по темам выполняемых работ	Владеет навыками подготовки и проведения публичных докладов по темам выполняемых работ
	ИДК ПК4.2 Владеет технологиями подготовки документов в том числе в областях математики и информатики, в том числе с использованием свободно-распространяемого программного обеспечения	Владеет технологиями подготовки документов для описание разработанного программного обеспечения
	ИДК ПК4.3 Способен проводить под научным руководством исследования на основе существующих методов в некоторой области профессиональной деятельности	Умеет проводить под научным руководством исследования на основе существующих методов в некоторой области профессиональной деятельности

4. СОДЕРЖАНИЕ И СТРУКТУРА ДИСЦИПЛИНЫ

Объем дисциплины составляет 7 зачетных единиц, 252 часа, в том числе 35 часа на контроль, практическая подготовка 128 часов.

Форма промежуточной аттестации: 1 семестр – зачёт, 2 семестр – экзамен.

4.1. СОДЕРЖАНИЕ ДИСЦИПЛИНЫ, СТРУКТУРИРОВАННОЕ ПО ТЕМАМ, С УКАЗАНИЕМ ВИДОВ УЧЕБНЫХ ЗАНЯТИЙ И ОТВЕДЕННОГО НА НИХ КОЛИЧЕСТВА АКАДЕМИЧЕСКИХ ЧАСОВ

№ п/п	Раздел дисциплины/темы	Семестр	Виды учебной работы, включая самостоятельную работу обучающихся и трудоемкость (в часах)				Формы текущего контроля успеваемости
			Контактная работа преподавателя с обучающимися			Самостоятельная работа + контроль	
			Лекции	Семинарские (практические занятия)	Контроль обучения		
1	Криптовалюты и смарт-контракты	1	10	12	2	30	тест
	Тема 1. Появление и развитие криптовалют						
	Тема 2. Технологии обеспечения доверия в децентрализованных распределенных сетях						
	Тема 3. Смарт-контракт — распределенные программы с верифицируемым исполнением						
2	Настройка рабочего окружения:	1	14	18	4	40	тест
	Тема 1. Системы контроля версий: локальные, централизованные распределенные						
	Тема 2. Системы контроля версий: GIT — Жизненный цикл состояний файлов						
	Тема 3. Системы контроля версий: GIT — тэги, ветвление и откат						
	Тема 4. Система управления пакетами: npm, yarn						
	Тема 5. Публичные системы распределенного хранения исходных						

№ п/п	Раздел дисциплины/темы	Семестр	Виды учебной работы, включая самостоятельную работу обучающихся и трудоемкость (в часах)				Формы текущего контроля успеваемости
			Контактная работа преподавателя с обучающимися			Самостоятельная работа + контроль	
			Лекции	Семинарские (практические занятия)	Контроль обучения		
	кодов GitHub, GitLab						
3	Разработка в сети Ethereum	2	22	26	4	68	Индивидуальные задания
	Тема 1. Solidity — основные синтаксические конструкции						
	Тема 2. Solidity — ограничения и особенности написания смарт-контрактов						
	Тема 3. Solidity — тестирование и отладка						
	Тема 4. Простые среды разработки: Remix						
	Тема 5. Сложные среды разработки: Truffle, Hardhat						
Итого часов			46	58	10	138	

4.2. План внеаудиторной самостоятельной работы обучающихся по дисциплине

Семестр	Название раздела, темы	Самостоятельная работа обучающихся			Оценочное средство	Учебно-методическое обеспечение самостоятельной работы
		Вид самостоятельной работы	Сроки выполнения	Затраты времени (час.)		
1	Системы контроля	<i>УИЛТ</i>	1-я половина курса + подготовка к аттест.	22	тест	УМО расположено в ИОС Educa на странице курса
1	Управление пакетами	<i>УИЛТ</i>	1-я половина курса+ подготовка к аттест.	26	тест	УМО расположено в ИОС Educa на странице курса
2	Синтаксические конструкции	<i>УИЛТ</i>	2-я половина курса+ подготовка к аттест.	30	тест	УМО расположено в ИОС Educa на странице курса
2	Тестирование и отладка	<i>УИЛТ</i>	2-я половина курса+ подготовка к аттест.	60	тест	УМО расположено в ИОС Educa на странице курса
Общая трудоемкость самостоятельной работы по дисциплине (час)				138		
Из них объем самостоятельной работы с использованием электронного обучения и дистанционных образовательных технологий (час)				138		

4.3. СОДЕРЖАНИЕ УЧЕБНОГО МАТЕРИАЛА

1. Криптовалюты и смарт-контракты

Тема 1. Появление и развитие криптовалют

Основные понятия, виды, типы, применение

Тема 2. Технологии обеспечения доверия в децентрализованных распределенных сетях

Основные виды технологий обеспечения доверия, особенности применения. Децентрализованные сети.

Тема 3. Смарт-контракт — распределенные программы с верифицируемым исполнением

Основные понятия, виды, применение смарт-контрактов. Методология работы со смарт-контрактами

2. Настройка рабочего окружения:

Тема 1. Системы контроля версий: локальные, централизованные распределенные

Виды контроля версий, основные принципы контроля.

Тема 2. Системы контроля версий: GIT — Жизненный цикл состояний файлов

Тема 3. Системы контроля версий: GIT — тэги, ветвление и откат

Тема 4. Система управления пакетами: npm, yarn

Тема 5. Публичные системы распределенного хранения исходных кодов GitHub, GitLab

3. Разработка в сети Ethereum

Тема 1. Solidity — основные синтаксические конструкции

Тема 2. Solidity — ограничения и особенности написания смарт-контрактов

Тема 3. Solidity — тестирование и отладка

Тема 4. Простые среды разработки: Remix

Тема 5. Сложные среды разработки: Truffle, Hardhat

4.3.1. Перечень семинарских, практических занятий и лабораторных работ

№ п/п	№ раздела и темы	Наименование семинаров, практических и лабораторных работ	Трудоемкость (час.)		Оценочные средства	Формируемые компетенции (индикаторы)*
			Всего часов	Из них практическая подготовка		
1	2	3	4	5	6	7
		Настройка и эксплуатация рабочей среды TS/JS приложений	10	10	устный опрос, компьютерные задания	ПК-2 (ИДК ПК2.1, ИДК ПК2.2, ИДК ПК2.3), ПК-4 (ИДК ПК4.1, ИДК ПК4.2, ИДК ПК4.3)
		Тестирование и отладка в node среде (mocha, chai)	12	12	устный опрос, компьютерные задания	ПК-2 (ИДК ПК2.1, ИДК ПК2.2, ИДК ПК2.3), ПК-4 (ИДК ПК4.1, ИДК ПК4.2, ИДК ПК4.3)
		Специфика исполнения, проектирования и реализации децентрализованных распределенных систем	12	12	устный опрос, компьютерные задания	ПК-2 (ИДК ПК2.1, ИДК ПК2.2, ИДК ПК2.3), ПК-4 (ИДК ПК4.1, ИДК ПК4.2, ИДК ПК4.3)
		Разработка, отладка и тестирование (solidity, hardhat)	14	14	устный опрос, компьютерные задания	ПК-2 (ИДК ПК2.1, ИДК ПК2.2, ИДК ПК2.3), ПК-4 (ИДК ПК4.1, ИДК ПК4.2, ИДК ПК4.3)
		Развертывание и интеграция смарт-контрактов	10	10	устный опрос, компьютерные задания	ПК-2 (ИДК ПК2.1, ИДК ПК2.2, ИДК ПК2.3), ПК-4 (ИДК ПК4.1, ИДК ПК4.2, ИДК ПК4.3)
		Всего	58	58		

4.3.2. Перечень тем (вопросов), выносимых на самостоятельное изучение студентами в рамках самостоятельной работы (СР)

Не предусмотрено

4.4. МЕТОДИЧЕСКИЕ УКАЗАНИЯ ПО ОРГАНИЗАЦИИ САМОСТОЯТЕЛЬНОЙ РАБОТЫ СТУДЕНТОВ

Методические указания по организации самостоятельной работы расположены в ИОС Edusa на странице курса

4.5. ПРИМЕРНАЯ ТЕМАТИКА КУРСОВЫХ РАБОТ (ПРОЕКТОВ)

- Децентрализованное распределенное приложение-игра:
 - Соревнование на знание ПДД
 - Экономическая игра “Рынок” (по мотивам игры “Ферма”)
- Автоматический распределитель призового вознаграждения, на основе статистики проводимых матчей Counter-Strike
- NFT-Криптогалерея

5. УЧЕБНО-МЕТОДИЧЕСКОЕ И ИНФОРМАЦИОННОЕ ОБЕСПЕЧЕНИЕ ДИСЦИПЛИНЫ (МОДУЛЯ)

а) перечень литературы

1. Крейдер, О. А. Информационные системы и технологии : учебное пособие / О. А. Крейдер. — Дубна : Государственный университет «Дубна», 2019. — 61 с. — ISBN 978-5-89847-577-2. — Текст : электронный // Лань : электронно-библиотечная система. — URL: <https://e.lanbook.com/book/154486> (дата обращения: 17.04.2022). — Режим доступа: для авториз. пользователей.
2. Орлов С.А. Программная инженерия. Технологии разработки программного обеспечения : учеб. для студ. вузов / С. А. Орлов. - 5-е изд., обновл. и доп. - СПб. : Питер, 2018. - 640 с. - ISBN 978-5-496-01917-0 (10 экз.)
3. Гаврилова, И. В. Разработка приложений [Электронный ресурс] : учеб. пособие / И. В. Гаврилова. - 3-е изд., стер. - Электрон. текстовые дан. - Москва : ФЛИНТА, 2017. - 243 с. ; есть. - Режим доступа: <https://lib.rucont.ru/efd/246530>. - ЭБС

б) базы данных, информационно-справочные и поисковые системы

1. git docs: сайт. — URL: <https://git-scm.com/> (дата обращения: 15.08.2022)
2. Solidity Programming Language : сайт. — URL: <https://soliditylang.org/> (дата обращения: 15.08.2022)
3. Node Packaged Modules : сайт. — URL: <https://www.npmjs.com/> (дата обращения: 15.08.2022)
4. Yarn package manager : сайт. — URL: <https://yarnpkg.com/> (дата обращения: 15.08.2023)
5. Ethereum development environment for professionals : сайт. — URL: <https://hardhat.org/> (дата обращения: 15.08.2022)
6. Chai Assertion Library : сайт. — URL: <https://www.chaijs.com/> (дата обращения: 15.08.2022)
7. Mocha - the fun, simple, flexible JavaScript test framework : сайт. — URL: <https://mochajs.org/> (дата обращения: 15.08.2022)

6. МАТЕРИАЛЬНО-ТЕХНИЧЕСКОЕ ОБЕСПЕЧЕНИЕ ДИСЦИПЛИНЫ

6.1. УЧЕБНО-ЛАБОРАТОРНОЕ ОБОРУДОВАНИЕ:

Компьютерный класс с выходом в интернет

6.2. ПРОГРАММНОЕ ОБЕСПЕЧЕНИЕ:

- *vs-code/sublime/webstorm (или аналог)*
- *GIT*
- *node (lts-18)*
- *npm*
- *yarn*
- *Yandex браузер (или любой другой, поддерживающий расширения chrome/mozilla)*
- *hardhat development tools*
- *TypeScript*

6.3. ТЕХНИЧЕСКИЕ И ЭЛЕКТРОННЫЕ СРЕДСТВА:

ИОС EDUCA, DOMIC, презентационное оборудование, персональный компьютер с возможностью демонстрации презентаций в формате pdf.

7. ОБРАЗОВАТЕЛЬНЫЕ ТЕХНОЛОГИИ

При реализации данного курса используются следующие образовательные технологии: технологии традиционного обучения, игровые технологии, технологии проблемного обучения, технологии обучения в сотрудничестве, технологии контекстного обучения, интерактивные технологии, технологии дистанционного обучения, активные педагогические технологии.

8. ОЦЕНОЧНЫЕ МАТЕРИАЛЫ ДЛЯ ТЕКУЩЕГО КОНТРОЛЯ И ПРОМЕЖУТОЧНОЙ АТТЕСТАЦИИ

8.1. ОЦЕНОЧНЫЕ МАТЕРИАЛЫ ДЛЯ ВХОДНОГО КОНТРОЛЯ

Не предусмотрено

8.2. ОЦЕНОЧНЫЕ МАТЕРИАЛЫ ТЕКУЩЕГО КОНТРОЛЯ

Задание 1.1

- Создать пустой пакет (npm/yarn)
- Реализовать программу, которая выводит в консоль количество времени оставшегося до ближайшего нового года
- Создать репозиторий на GitHub
- Опубликовать исходные коды на внешнем репозитории (GitHub)

Дополнительно:

- Учесть текущую таймзону сервера
- Реализовать программу с использованием языка TS

Задание 1.2

- В отдельной ветке написать TS функцию, которая принимает дату и считает время оставшееся от этой даты до нового года
- Из main файла вызвать функцию с new Date() в качестве параметра
- Закоммитить изменения в отдельной ветке
- Слить новую ветвь в мастер-ветку
- Опубликовать в своем репозитории

Задание 1.3

- Вынести функцию в отдельный файл *.ts и экспортировать ее
- Импортировать обозначенную выше функцию и вызвать ее из index.ts
- Слить ветки, опубликовать на внешнем репозитории

Задание 2

- Ознакомится с документацией библиотек **mocha**, **chai**
- Согласно документации написать набор unit-тестов для функции из Задания 1

Задание 3

- Произвести форк (клонирование) репозитория mpt_4
- В папке packages создать свою директорию, имя которой соответствует вашему логину на github
- Внутри своей папки инициализировать npm пакет работа должна вестись через yarn или npm, если yarn не работает локально
- Перенести свой предыдущий проект в этот модуль
- Опубликовать изменения в свой репозиторий
- Запросить пул реквест

Дополнительно:

- Сгенерировать собственный ssh-ключ, подключить к github, настроить работу через него
- Сгенерировать собственный gpg-ключ rsa-rsa 4096, подключить к github, настроить подпись коммитов

8.3. ОЦЕНОЧНЫЕ МАТЕРИАЛЫ ПРОМЕЖУТОЧНОГО КОНТРОЛЯ

Экзамен принимается в форме защиты группового проекта. Группа может состоять из 3~6 человек в зависимости от сложности выбранной задачи. Согласование тем проектов и распределение участников производится на отдельно выделенном занятии. Не позднее, чем за 3 дня до экзамена, каждому студенту следует сформировать персональный отчет, в котором должны быть отражены персональная область ответственности, перечень лично выполненных задач, краткое текстовое описание о своей роли в команде. в день экзамена, каждая группа презентует свой проект, демонстрируя выполнение всех заявленных требований.

Требования к проекту:


- **Работоспособность.** Проект должен запускаться и выполнять свою базовую заявленную функциональность
- **Документированность.** Бизнес-логика, крупные функциональные блоки и их взаимодействие, а также внутреннее содержание таких блоков (диаграммы классов/контрактов/сущностей и т.д.) должны быть задокументировано, с использованием тех инструментов, которыми студенты овладели ранее. Также должна быть документация описывающая действия необходимые для того, чтобы развернуть все компоненты приложения. Код должен быть опубликован на внешней репозитории, доступ к которому должен быть предоставлен не позднее чем за 1 день до даты экзамена.
- **Тестирование.** Покрытие кода контрактов тестами должно составлять не менее 75%
- **Содержание.** Проект должен включать в себя по меньшей мере 2 крупных модуля:
 - Контракты, содержащие основную бизнес-логику
 - Клиент, выполненный либо в виде web-интерфейса, либо в виде бота взаимодействующего с контрактами

Проект оценивается следующим образом:

- В случае выполнения всех требований проект оценивается на *отлично*.
- Если полностью не выполнено какое-то одно из требований, или несколько требований выполнены не в полном объеме — *хорошо*.
- Если полностью не выполнено 2 требования — *удовлетворительно*.
- Иначе, если не выполнены более 3-х требований — *неудовлетворительно*

Оценки участникам распределяются на основании личных отчетов. По-умолчанию считается, что вклад всех участников команды равноценен, и оценка каждого студента соответствует оценке проекта, в котором он принимал участие. В случае, если из совокупности личных отчетов следует, что вклад конкретного студента несоизмеримо мал, то оценка этого студента может быть снижена.

Разработчики:



(подпись)

ассистент
(занимаемая должность)

Гаврилин Д. Н.
(Ф.И.О.)

Программа составлена в соответствии с требованиями ФГОС ВО по направлению подготовки 09.04.03 «Прикладная информатика» (уровень магистратуры), утвержденный приказом Министерства образования и науки Российской Федерации от «19» сентября 2017 г. № 916, зарегистрированный в Минюсте России «10» октября 2017 г. № 48495 с изменениями и дополнениями от 26.11.2020 г., 8.02.2021 г.

Программа рассмотрена на заседании кафедры Алгебраических и информационных систем ИМИТ ИГУ «24» марта 2022 г.

Протокол № 9 Зав. кафедрой _____  _____ Пантелеев В.И.

Настоящая программа, не может быть воспроизведена ни в какой форме без предварительного письменного разрешения кафедры-разработчика программы.