

**МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ,  
МИНИСТЕРСТВО НАУКИ, ВЫСШЕГО ОБРАЗОВАНИЯ И ИННОВАЦИЙ  
КЫРГЫЗСКОЙ РЕСПУБЛИКИ**

МОО ВО Кыргызско-Российский Славянский университет  
имени первого Президента Российской Федерации Б.Н. Ельцина



## Проектирование распределенных информационных систем

### рабочая программа дисциплины (модуля)

Закреплена за кафедрой	<b>Информационных и вычислительных технологий</b>
Учебный план	g090404_24_12пи_рпис.plx Направление подготовки 09.04.04 - РФ, 710400 - КР Программная инженерия Магистерская программа "Разработка программно-информационных систем"
Квалификация	<b>магистр</b>
Форма обучения	<b>очная</b>
Общая трудоемкость	<b>4 ЗЕТ</b>
Часов по учебному плану	144
в том числе:	
аудиторные занятия	38
самостоятельная работа	103,8
	Виды контроля в семестрах: зачет с оценкой 2 курсовая работа 2

#### Распределение часов дисциплины по семестрам

Семестр (<Курс>.<Семестр на курсе>)	2 (1.2)		Итого	
	уп	рп	уп	рп
Неделя	18			
Вид занятий				
Лекции	16	16	16	16
Практические	22	22	22	22
Контактная работа в период теоретического обучения	2,2	2,2	2,2	2,2
В том числе инт.	8	8	8	8
В том числе в форме практ.подготовки	22	22	22	22
Итого ауд.	38	38	38	38
Контактная работа	40,2	40,2	40,2	40,2
Сам. работа	103,8	103,8	103,8	103,8
Итого	144	144	144	144

Программу составил(и):

ст. преп. , *Беляев А.А.*; к.т.н. , доцент, *Хмелева И.В.*



Рабочая программа дисциплины

разработана в соответствии с ФГОС 3++:

Федеральный государственный образовательный стандарт высшего образования - магистратура по направлению подготовки 09.04.04 Программная инженерия (приказ Минобрнауки России от 19.09.2017 г. № 932)

составлена на основании учебного плана:

Направление подготовки 09.04.04 - РФ, 710400 - КР Программная инженерия

Магистерская программа "Разработка программно-информационных систем"

утвержденного учёным советом вуза от 22.10.24 протокол № 2

Рабочая программа одобрена на заседании кафедры

Протокол от 03.09.2025 г. № 1

Срок действия программы: 2024-2028 уч.г.

Зав. кафедрой д.т.н., проф. Лыченко Н.М.



---

---

**Визирование РПД для исполнения в очередном учебном году**

Председатель УМС

\_\_\_\_\_ 2026 г.

Рабочая программа пересмотрена, обсуждена и одобрена для исполнения в 2026-2027 учебном году на заседании кафедры

Протокол от \_\_\_\_\_ 2026 г. № \_\_\_\_  
Зав. кафедрой д.т.н., проф. Лыченко Н.М.

---

---

**Визирование РПД для исполнения в очередном учебном году**

Председатель УМС

\_\_\_\_\_ 2027 г.

Рабочая программа пересмотрена, обсуждена и одобрена для исполнения в 2027-2028 учебном году на заседании кафедры

Протокол от \_\_\_\_\_ 2027 г. № \_\_\_\_  
Зав. кафедрой д.т.н., проф. Лыченко Н.М.

---

---

**Визирование РПД для исполнения в очередном учебном году**

Председатель УМС

\_\_\_\_\_ 2028 г.

Рабочая программа пересмотрена, обсуждена и одобрена для исполнения в 2028-2029 учебном году на заседании кафедры

Протокол от \_\_\_\_\_ 2028 г. № \_\_\_\_  
Зав. кафедрой д.т.н., проф. Лыченко Н.М.

---

---

**Визирование РПД для исполнения в очередном учебном году**

Председатель УМС

\_\_\_\_\_ 2029 г.

Рабочая программа пересмотрена, обсуждена и одобрена для исполнения в 2029-2030 учебном году на заседании кафедры

Протокол от \_\_\_\_\_ 2029 г. № \_\_\_\_  
Зав. кафедрой д.т.н., проф. Лыченко Н.М.

**1. ЦЕЛИ ОСВОЕНИЯ ДИСЦИПЛИНЫ**

1.1	Целью изучения дисциплины «Распределенные информационные системы» является подготовка специалистов, обладающих фундаментальными знаниями и практическими навыками в области построения распределенных информационных систем и сетей, программной инженерии, теории и практики руководства проектами по созданию распределенных информационных систем.
-----	---

**2. МЕСТО ДИСЦИПЛИНЫ В СТРУКТУРЕ ООП**

Цикл (раздел) ООП:		Б1.В
<b>2.1</b>	<b>Требования к предварительной подготовке обучающегося:</b>	
2.1.1	Дисциплины ООП бакалавриата по направлению 09.03.04 Программная инженерия"	
<b>2.2</b>	<b>Дисциплины и практики, для которых освоение данной дисциплины (модуля) необходимо как предшествующее:</b>	
2.2.1	Тестирование и обеспечение качества программных систем, Методы сопровождения ПО, Проектирование высоконагруженных систем, Преддипломная практика	

**3. КОМПЕТЕНЦИИ ОБУЧАЮЩЕГОСЯ, ФОРМИРУЕМЫЕ В РЕЗУЛЬТАТЕ ОСВОЕНИЯ ДИСЦИПЛИНЫ (МОДУЛЯ)****ПК-3: Владение методами программной реализации распределенных информационных систем**

<b>Знать:</b>	
Уровень 1	основные технологии построения распределенных систем: процедурные подходы, объектные и компонентные модели; базовые шаблоны проектирования распределенных систем.
<b>Уметь:</b>	
Уровень 1	применить изученные методы и алгоритмы на практике в процессе разработки реальных распределенных программных систем
<b>Владеть:</b>	
Уровень 1	практическими навыками работы с современными системами проектирования и разработки распределенных систем

**В результате освоения дисциплины обучающийся должен**

<b>3.1</b>	<b>Знать:</b>
3.1.1	- базовые принципы построения распределенных систем: синхронизация времени и логические часы, распределенные транзакции, взаимодействие по сети и Удаленный Вызов Процедур.
<b>3.2</b>	<b>Уметь:</b>
3.2.1	- применять механизмы и средства современных операционных систем, используемых для построения распределенных приложений,
3.2.2	- разрабатывать приложения в Map Reduce,
3.2.3	- создавать сетевые файловые системы.
<b>3.3</b>	<b>Владеть:</b>
3.3.1	навыками применения инструментальных программных средств, применяемые для построения распределенных систем.

**4. СТРУКТУРА И СОДЕРЖАНИЕ ДИСЦИПЛИНЫ (МОДУЛЯ)**

Код занятия	Наименование разделов и тем /вид занятия/	Семестр / Курс	Часов	Компетенции	Литература	Инте ракт.	Пр. подг.	Примечание
	<b>Раздел 1. Базовые знания по распределенным системам</b>							
1.1	Распределенные систмы. Map Reduce. Сетевые файловые системы. Взаимодействие по сети и Удаленный Вызов Процедур. Синхронизация времени и логические часы. /Лек/	2	2			2		
1.2	Удаленный вызов процедур и потоки /Пр/	2	2				2	
1.3	Echo Server /Ср/	2	18					
	<b>Раздел 2. Отказоустойчивость</b>							

2.1	Первичный бэкап. Двух-фазовый комит, введение в безопасность и отказоустойчивость /Лек/	2	2					
2.2	Согласованность данных : FLP невозможность, Paxos, Реплицируемые машины состояний, Raft. Отказоустойчивость Byzantine /Лек/	2	2					
2.3	Первичный бэкап. Двух-фазовый комит. Реплицируемые машины состояний, Raft /Пр/	2	6			2	6	
2.4	Map Reduce /Ср/	2	16					
	<b>Раздел 3. Масштабируемость, согласованность данных и транзакции</b>							
3.1	Пиринговые системы и распределенные хэш таблицы. Возможность согласованности данных. Масштабируемые сервисы: Key-Value хранилища. /Лек/	2	2					
3.2	Масштабируемые сервисы: Key-Value хранилища. /Пр/	2	6				6	
3.3	Сильная согласованность данных и теорема CAP. Причинная согласованность. Контроль конкурентности, блокировки, и восстановление, (OCC, MVCC) и распределенные транзакции. Разрешение конфликтов, Криптография, Недоверенные облачные сервисы /Лек/	2	2					
3.4	Raft /Ср/	2	24					
3.5	Контроль конкурентности, блокировки, и восстановление, (OCC, MVCC) и распределенные транзакции. Разрешение конфликтов /Пр/	2	4				4	
3.6	Отказоустойчивый Key/Value сервис /Ср/	2	24					
3.7	Blockchains. Сети дотавки контента. Distributed Mesh Wireless Networks. /Лек/	2	2			2		
	<b>Раздел 4. Обработка больших данных</b>							
4.1	Graph processing. Обработка потоков. Управление расписанием кластеров /Лек/	2	4					
4.2	Обработка потоков. Управление расписанием кластеров /Пр/	2	4			2	4	
4.3	Шардированный Key/Value сервис /Ср/	2	21,8					
4.4	Обработка потоков /КрТО/	2	2,2					

## 5. ФОНД ОЦЕНОЧНЫХ СРЕДСТВ

### 5.1. Контрольные вопросы и задания

### 5.2. Темы курсовых работ (проектов)

1. Projects with a real-time multi-user component are a good match for Paxos - you could either have the individual users be part of the Paxos quorum, or use Paxos for the replicated state that stores their updates.
2. Shared document editing, in the style of Google docs. The system should support real-time editing and viewing by multiple participants. Multiple replicas would be maintained for fault tolerance. Caching and/or copy migration would be useful to minimize application response time.
3. A simulated life game, in the style of The Sims or Farmville. The state of the system would be partitioned spatially, with replication for fault tolerance.
4. A multi-player real-time game - there are many possibilities here.
5. An airline reservation system. Each airline would maintain its own collection of servers, with enough state replication to enable automatic fail-over. It would be possible to book travel that involves multiple airlines.
6. A low-latency notification system. E.g., watch a whole bunch of RSS feeds and send all subscribers an email when one is updated. Interface with both the raw RSS feeds and Google's update notification service. Replicate and partition the state of the monitoring system so that it can scale and survive node failures.
7. Projects involving multiple agents or users coordinating updates that can be shown on a map (e.g., Google Maps). Integrate this with Google Maps and some AJAX - makes for a good easy GUI.
8. Ambitious projects in the past have involved creating Android/iOS smart phone apps and games.
9. Implement a distributed file system that does something interesting. Maybe you want one for storing your MP3s or movies. Or perhaps for something entirely different. What design constraints and optimization criteria would you have?
10. A solution to a problem that really bugs you or that you would like to see solved - perhaps your new startup idea!

### 5.3. Фонд оценочных средств

### 5.4. Перечень видов оценочных средств

## 6. УЧЕБНО-МЕТОДИЧЕСКОЕ И ИНФОРМАЦИОННОЕ ОБЕСПЕЧЕНИЕ ДИСЦИПЛИНЫ (МОДУЛЯ)

### 6.1. Рекомендуемая литература

### 6.3. Перечень информационных и образовательных технологий

#### 6.3.1 Компетентностно-ориентированные образовательные технологии

#### 6.3.2 Перечень информационных справочных систем и программного обеспечения

## 7. МАТЕРИАЛЬНО-ТЕХНИЧЕСКОЕ ОБЕСПЕЧЕНИЕ ДИСЦИПЛИНЫ (МОДУЛЯ)

## 8. МЕТОДИЧЕСКИЕ УКАЗАНИЯ ДЛЯ ОБУЧАЮЩИХСЯ ПО ОСВОЕНИЮ ДИСЦИПЛИНЫ (МОДУЛЯ)