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

УТВЕРЖДАЮ
и.о. декана факультета ПИФ
_____ Н. М. Комаров
22.09.2025

Спецкурс по разработке программно-информационных систем

рабочая программа дисциплины (модуля)

Закреплена за кафедрой **Информационных и вычислительных технологий**

Учебный план g090404_25_1пи_рпис.plx
Направление подготовки 09.04.04 - РФ, 710400 - КР Программная инженерия
Магистерская программа "Разработка программно-информационных систем"

Квалификация **магистр**

Форма обучения **очная**

Общая трудоемкость **3 ЗЕТ**

Часов по учебному плану 108
в том числе:
аудиторные занятия 28
самостоятельная работа 79,9

Виды контроля в семестрах:
зачет 3

Распределение часов дисциплины по семестрам

Семестр (<Курс>.<Семестр на курсе>)	**3 (2.1)**		Итого	
Недель	18			
Вид занятий	УП	РП	УП	РП
Лекции	12	12	12	12
Практические	16	16	16	16
Контактная работа в период теоретического обучения	0,1	0,1	0,1	0,1
Итого ауд.	28	28	28	28
Контактная работа	28,1	28,1	28,1	28,1
Сам. работа	79,9	79,9	79,9	79,9
Итого	108	108	108	108

Программу составил(и):
к.т.н., доцент, Хмелева И.В.; к.т.н. , доцент, Манжикова С.Ц. _____

д.т.н., проф. , Верзунов С.Н. _____

Рабочая программа дисциплины
Спецкурс по разработке программно-информационных систем

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 30.06.2025 протокол № 13

Рабочая программа одобрена на заседании кафедры
Информационных и вычислительных технологий

Протокол от 03.09.2025 г. № 1
Срок действия программы: 2025-2027 уч.г.
Зав. кафедрой д.т.н., проф. Лыченко Н.М.

ЛИСТ РЕГИСТРАЦИИ ИЗМЕНЕНИЙ
к рабочей программе дисциплины (модуля)

Спецкурс по разработке программно-информационных систем

наименование дисциплины

№ п/п	Основание внесения изменения	Раздел/ пункт	Содержание изменения
1.	№357-П от 24 июня 2026- О преобразовании Естественно-технического факультета и Факультета архитектуры дизайна и строительства в Политехнический институт КРСУ	Титульный лист. Пункт Разработано	Заменить слова «Разработано Кафедра Информационных и вычислительных технологий» словами «Разработано: Высшая школа Цифровых технологий»
2.			
3.			
4.			
5.			

1. ЦЕЛИ ОСВОЕНИЯ ДИСЦИПЛИНЫ	
1.1	Теоретическая и практическая подготовка студентов в области информационных технологий в такой степени, чтобы они могли выбирать необходимые технические, программные и технологические решения, уметь объяснить принципы их функционирования и правильно их использовать.
1.2	Иметь представление о каждом этапе жизненного цикла программы – от проектирования до внедрения и сопровождения. Знать современные перспективные направления развития технологии разработки ПО.
1.3	Ознакомление с программными средствами, используемые в индустриальном производстве ПО.

2. МЕСТО ДИСЦИПЛИНЫ В СТРУКТУРЕ ООП	
Цикл (раздел) ООП:	ФТД
2.1	**Требования к предварительной подготовке обучающегося:**
2.1.1	Информационная безопасность открытых систем
2.1.2	Облачные инфраструктуры и сервисы
2.1.3	Проектирование высоконагруженных систем
2.1.4	Тестирование и обеспечение качества программных средств
2.1.5	Методология программной инженерии
2.1.6	Технологическая (проектно-технологическая) практика
2.1.7	Принципы WEB - программирования
2.1.8	Проектирование распределенных информационных систем
2.2	**Дисциплины и практики, для которых освоение данной дисциплины (модуля) необходимо как предшествующее:**
2.2.1	Подготовка к защите и защита выпускной квалификационной работы
2.2.2	Научно-исследовательская работа
2.2.3	Преддипломная практика
2.2.4	Выполнение и защита выпускной квалификационной работы

3. КОМПЕТЕНЦИИ ОБУЧАЮЩЕГОСЯ, ФОРМИРУЕМЫЕ В РЕЗУЛЬТАТЕ ОСВОЕНИЯ ДИСЦИПЛИНЫ (МОДУЛЯ)	
ОПК-8: Способен осуществлять эффективное управление разработкой программных средств и проектов.	
Знать:	
Уровень 1	- научные основы методов решения конкретных задач по модификации действующих ПС; - метрические системы оценки качества и сравнительного анализа модифицированных ПС.
Уметь:	
Уровень 1	- практически применять методы сопровождения и эволюции используемого ПО; - формулировать конкретные задачи, решение которых позволяет усовершенствовать и/или частично изменить функционал работающего ПО.
Владеть:	
Уровень 1	- навыками и инструментами анализа функционирующих ПС; - средствами планирования работ по адаптации и модификации ПС в конкретных условиях эксплуатации.

В результате освоения дисциплины обучающийся должен

3.1	**Знать:**
3.1.1	жизненный цикл программ;
3.1.2	- процессы планирования и управления проектами;
3.1.3	- методы оценки задач при реализации проекта;
3.1.4	- современные методологии разработки ПО (Agile, SCRUM, Kanban, XP и т.д.);
3.1.5	- особенности работы в команде, версионирование (Git, SVN, Github, Bitbucket, Jira).
3.2	**Уметь:**
3.2.1	применять методологии разработки;
3.2.2	осуществлять выбор программных и инструментальных средств для разработки;
3.2.3	- оценивать трудозатраты задач при реализации проекта;
3.2.4	- организовать работу команды разработки ПО;
3.2.5	- настраивать CI/DI;
3.2.6	- настраивать Workflow в Jira/TFS.
3.3	**Владеть:**

3.3.1	навыками работы в:
3.3.2	- Jira software, Confluence, messengers (Slack, Skype и т. д.);
3.3.3	- Team Foundation Server;
3.3.4	- Git, SVN, SmartGit;
3.3.5	- Github, Bitbucket.

4. СТРУКТУРА И СОДЕРЖАНИЕ ДИСЦИПЛИНЫ (МОДУЛЯ)

Код занятия	Наименование разделов и тем /вид занятия/	Семестр / Курс	Часов	Компетен-ции	Литература	Инте ракт.	Пр. подг.	Примечание
	Раздел 1. Введение в дисциплину. Методологии разработки ПО							
1.1	Основные понятия дисциплины и современные программные средства используемые при разработке ПО; Agile — общая методология гибкой разработки ПО. Гибкая методология - Scrum XP (Extreme Programming) — экстремальная разработка в динамической среде. /Лек/	3	2	ОПК-8	Л1.1Л2.1 Э1 Э2 Э3			
1.2	Гибкая методология – Kanban Сравнение гибких методологий Scrum и Kanban /Лек/	3	2	ОПК-8	Л1.3 Л1.1Л2.2 Э1 Э2 Э3			
1.3	Оценка задач в гибких методологиях Scrum и Kanban /Лек/	3	2	ОПК-8	Л1.1Л2.1 Э3			
1.4	Практическая работа №1. Создание проекта и описание задач в Team Foundation Server /Пр/	3	4	ОПК-8	Л1.3 Л1.1Л2.1 Э2 Э3			
1.5	Виды программирования. Подробное рассмотрение методологий разработки программного обеспечения - Каскадная, водопадная модель; - V-модель; - Итеративная и Инкрементная модель; - Спиральная модель; - Rapid Application Development; /Ср/	3	20,8	ОПК-8	Л1.1 Л1.3Л2.1 Э1 Э2 Э3			
1.6	Проработка лекционного материала /Ср/	3	14,1	ОПК-8	Э3			
	Раздел 2. Программные средства используемые при разработке ПО							
2.1	Team Foundation Server /Лек/	3	2	ОПК-8	Л1.2 Л1.1 Э3			
2.2	Jira software, Confluence, messengers (Slack, Skype и т. д.);Git, SVN, Github, Bitbucket, SmartGit /Лек/	3	2	ОПК-8	Л1.3 Л1.1Л2.2 Э2 Э3			
2.3	Практическая работа №2. SmartGit + Github создание проектов в команде /Пр/	3	4	ОПК-8	Л1.3 Л1.1 Э3			

2.4	Практическая работа №3. Создание Workflow для Kanban в Jira. Создание проекта и описание задач в Jira /Пр/	3	4	ОПК-8	Л1.3 Л1.1 Э2 Э3			
2.5	Проработка лекционного материала. Подготовка к практическим занятиям /Ср/	3	20	ОПК-8	Л1.1 Э3			
	Раздел 3. Раздел 3. Развертывание информационных систем							
3.1	CI/CD. Основные понятия. Инструменты для настройки CI/DI. /Лек/	3	2	ОПК-8	Л1.2 Л1.1Л2.2 Э2 Э3			
3.2	Практическая работа №4. Настройка CI в Team Foundation Server /Пр/	3	4	ОПК-8	Л1.1Л2.1 Э3			
3.3	Проработка лекционного материала. Подготовка к практическим занятиям /Ср/	3	25	ОПК-8	Л1.1 Л1.3Л2.1 Л2.2 Э3			
3.4	Зачет /КрТО/	3	0,1	ОПК-8	Л1.2 Л1.1 Э2 Э3			

5. ФОНД ОЦЕНОЧНЫХ СРЕДСТВ

5.1. Контрольные вопросы и задания

Контрольные вопросы для проверки ЗНАТЬ:

1. Основные процессы разработки ПО.
2. Роли в команде разработке ПО.
3. Виды программирования.
4. Методологии разработки программного обеспечения. Каскадная, водопадная модель. V-модель. Итеративная и Инкрементная модель. Спиральная модель. Rapid Application Development.
5. Agile – общая методология гибкой разработки.
6. Scrum. XP (Extreme Programming).
7. Kanban.
8. Методы оценки задач в гибких методологиях Scrum и Kanban.
9. Основные понятия Git, SVN, Github, Bitbucket, SmartGit.
10. Основные понятия Jira software, Confluence.
11. CI/CD. Основные понятия. Инструменты для настройки CI/DI.

Контрольные задачи для проверки УМЕТЬ:

1. Создать проект и описать задачи в Team Foundation Server/Jira.
2. Настроить Workflow в Team Foundation Server/Jira.
3. Настройка CI в Team Foundation Server.

Контрольные задания для проверки ВЛАДЕТЬ:

1. Использование Team Foundation Server/Jira для ведения проекта.
2. Создание проектов в команде с применением SmartGit + Github.
3. Применение CI/CD для проектов.

5.2. Темы курсовых работ (проектов)

нет

5.3. Фонд оценочных средств

Практическая работа №1.

1. Создать проект в среде Team Foundation Server.
2. Выбрать любую тему проекта.
3. Создать пользователей для команды проекта.
4. Настроить роли для пользователей.
5. Составить перечень всех Features для реализации проекта.
6. Провести декомпозицию Features на User Story.
7. Создать несколько Sprint и запланировать их.
8. Провести Estimate для User Story в первом Sprint.
9. Ознакомится с отчетами, которые предоставляет Team Foundation Server.

10. Эмулировать процесс работы над проектом.
11. Проследить динамику в диаграмме Burndown Chart в процессе работы.

Практическая работа №2.
1. Создать репозиторий на площадке Github и Bitbacket.
2. Включить пользователей в проект.
3. Создать проекты в каждом из репозиториев.
4. Выполнить основные команды Git (Commit, Pull, Push, Stage и т.д.)
5. Эмулировать командную работу над проектом.

Практическая работа №3.
1. Создать проект в среде Jira.
2. Выбрать любую тему проекта.
3. Создать пользователей для команды проекта.
4. Настроить роли для пользователей.
5. Настроить Workflow со столбцами Backlog, To do, Blocked, In Progress, In review, In Testing, Done.
6. Составить перечень всех User Story для реализации проекта.
7. Провести декомпозицию User Story на Task.
8. Создать несколько Sprint и запланировать их.
9. Провести Estimate для User Story в первом Sprint.
10. Создать документацию по проекту в среде Confluence.

Практическая работа №4.
1. Развернуть build agent на сервере.
2. Настроить build для проекта из Практической работы №2.
3. Развернуть deploy agent на сервере.
4. Настроить deploy проекта на сервер.

5.4. Перечень видов оценочных средств

Практическая работа;
Виды шкал оценивания представлены в Приложении 1.

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

6.1. Рекомендуемая литература

6.1.1. Основная литература

	Авторы, составители	Заглавие	Издательство, год
Л1.1	А.В.Панов	Разработка управленческих решений: информационные технологии: Учебное пособие для вузов	М.: Горячая линия- Телеком 2012
Л1.2	А.А. Малюк, В.С. Горбатов, В.И. Королев и др.	Введение в информационную безопасность: Учебное пособие для вузов	М.: Горячая линия- Телеком 2013
Л1.3	В. Г. Олифер, Н.А. Олифер.	Компьютерные сети. Принципы, технологии, протоколы: Учебник для ВУЗов	СПб.: Питер 2012

6.1.2. Дополнительная литература

	Авторы, составители	Заглавие	Издательство, год
Л2.1	Сошников Д	Функциональное программирование на F#	М., ДМК Пресс 2011
Л2.2	Э. Таненбаум	Компьютерные сети: Учебное пособие	СПб.: Питер 2007

6.2. Перечень ресурсов информационно-телекоммуникационной сети "Интернет"

Э1	Центр разработки Windows Azure	http://msdn.microsoft.com/windowsazure/
Э2	Pattern Windows Azure Design Pattern Catalog	http://neudesic.blob.core.windows.net/azuredesignpatt
Э3	Клементьев И.П. Устинов В.А. Введение в Облачные вычисления	http://www.twirpx.com/file/452598/

6.3. Перечень информационных и образовательных технологий

6.3.1 Компетентностно-ориентированные образовательные технологии

6.3.1.1	Изучение дисциплины студентами осуществляется в форме лекций, практических занятий в аудиторных условиях (компьютерные классы) и в процессе самостоятельной работы, контроля знаний.
6.3.1.2	Теоретическая информация представляется в виде компьютерных презентаций с использованием мультимедийных средств.

6.3.1.3	Практические занятия проводятся в компьютерных классах, оснащенных персональными компьютерами с необходимыми параметрами и с установленным необходимым программным обеспечением. Используется Интернет для получения дополнительной информации. Используется дискуссионный метод проведения занятий, где студенты могут высказать свое мнение по обсуждаемой проблеме.
6.3.1.4	Защита практических работ проводится в виде собеседования с преподавателем по теории и программной реализации работы.
6.3.2 Перечень информационных справочных систем и программного обеспечения	
6.3.2.1	Team Foundation Server;
6.3.2.2	Jira. Confluence;
6.3.2.3	Github;
6.3.2.4	Bitbacket
6.3.2.5	SmartGit

7. МАТЕРИАЛЬНО-ТЕХНИЧЕСКОЕ ОБЕСПЕЧЕНИЕ ДИСЦИПЛИНЫ (МОДУЛЯ)	
7.1	Учебная лаборатория программно-технического обеспечения:
7.2	ПК - 10 шт;
7.3	сервер - 1;
7.4	ПК преподавателя - 1.
7.5	Локальная сеть кафедры ИВТ КРСУ.
7.6	Интернет со скоростью 70 Мбит/сек.
7.7	Зона WI-FI.
7.8	Интерактивная доска, проектор, обычная доска, 20 посадочных мест

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

1. Технологическая карта дисциплины представлена в Приложении 2.

2. Рубежный контроль осуществляется в течение семестра в виде защиты практических работ.

Методические указания по выполнению практических работ представлены в электронной папке преподавателя (локальная сеть кафедры Информационных и вычислительных технологий КРСУ).
Подготовка к практическим работам предполагает самостоятельную работу студента по овладению навыками практического применения лекционного материала.