

Министерство науки и высшего образования Российской Федерации
Федеральное государственное автономное образовательное учреждение
высшего образования «Новосибирский национальный исследовательский
государственный университет» (Новосибирский государственный университет, НГУ)

Факультет информационных технологий

СОГЛАСОВАНО

Декан ФИТ НГУ


М.М. Лаврентьев

«23» июля 2020 г.

РАБОЧАЯ ПРОГРАММА ДИСЦИПЛИНЫ

Управление производственным процессом разработки программного обеспечения

Направление подготовки: 09.03.01 ИНФОРМАТИКА И ВЫЧИСЛИТЕЛЬНАЯ ТЕХНИКА

Направленность (профиль): Программная инженерия и компьютерные науки

Форма обучения: очная

Год обучения: 3, семестр: 6

№	Вид деятельности	Семестр
		6
1	Лекции, час.	32
2	Практические занятия, час.	32
3	Лабораторные занятия, час.	
4	Занятий в контактной форме без учета промежуточной аттестации, час, из них	66
5	в электронной форме, час.	
6	из них аудиторных занятий, час.	64
7	из них в активной и интерактивной форме, час.	32
8	консультаций, час.	2
9	Самостоятельная работа, час.	40
10	в том числе на выполнение письменных работ, час	
11	Форма аттестации (экзамен, зачет, дифференцированный зачет), час	ДЗ 2
12	Всего зачетных единиц ¹	3

Новосибирск 2020

¹ С учетом выделенных часов на промежуточную аттестацию

Рабочая программа дисциплины составлена на основании федерального государственного образовательного стандарта (ФГОС) высшего образования - бакалавриат по направлению подготовки 09.03.01 ИНФОРМАТИКА И ВЫЧИСЛИТЕЛЬНАЯ ТЕХНИКА.

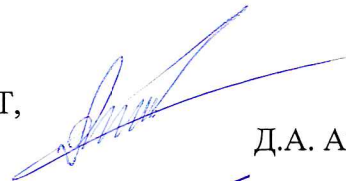
Федеральный государственный образовательный стандарт (ФГОС) высшего образования - бакалавриат по направлению подготовки 09.03.01 ИНФОРМАТИКА И ВЫЧИСЛИТЕЛЬНАЯ ТЕХНИКА введен в действие приказом Минобрнауки от 19.09.2017 № 929.

Место дисциплины в структуре учебного плана: Блок 1 Дисциплины (модули), обязательная часть, обязательная дисциплина.

Рабочая программа дисциплины утверждена решением Ученого совета факультета информационных технологий от 22.07.2020, протокол № 77.

Программу разработал:

ст. преподаватель кафедры общей информатики ФИТ,
магистр математики



Д.А. Анойкин

Заведующий кафедрой общей информатики ФИТ,
доктор физико-математических наук



Д.Е. Пальчунов

Ответственный за образовательную программу:

доцент кафедры систем информатики ФИТ,
кандидат технических наук



А.А. Романенко

Аннотация к рабочей программе дисциплины «Управление производственным процессом разработки программного обеспечения»

Дисциплина «Управление производственным процессом разработки программного обеспечения» реализуется в рамках образовательной программы высшего образования – программы бакалавриата 09.03.01 ИНФОРМАТИКА И ВЫЧИСЛИТЕЛЬНАЯ ТЕХНИКА, направленность (профиль): ПРОГРАММНАЯ ИНЖЕНЕРИЯ И КОМПЬЮТЕРНЫЕ НАУКИ по очной форме обучения на русском языке.

Дисциплина «Управление производственным процессом разработки программного обеспечения» имеет своей целью:

- ознакомление студентов с современными инструментами управления процессом разработки программного обеспечения.
- ознакомление студентов с примерами использования инструментов управления разработкой программного обеспечения. В качестве основных примеров используются проекты разработки программного обеспечения с открытым программным кодом.
- получение будущим выпускником-бакалавром первого практического навыка коллективного выполнения проекта по разработке программного обеспечения, используя современные инструменты управления процессом разработки.
- получение практического навыка в работе с проектной документацией, средствами контроля версий, планирования потоков работ, управления задачами и управления дефектами.

Данный курс позволяет студентам изучить современные инструменты управления разработкой программного обеспечения, без которых невозможно представить разработку даже небольшого проекта. Студенты, прослушавшие курс, не только смогут быстро войти в команду разработчиков, но и предложить более эффективные методы и инструменты организации процесса разработки.

Уникальность курса заключается в соединении практических занятий магистрантов, проходящих курс «Практика управления ИТ-проектами», и бакалавров данного курса: магистранты участвуют в учебном проекте в роли менеджеров, архитекторов, специалистов по качеству, бакалавры – в ролях разработчиков, тестировщиков, дизайнеров пользовательского интерфейса и т.д.

Место в образовательной программе

Дисциплина «Управление производственным процессом разработки программного обеспечения» развивает знания, умения и навыки, сформированные у обучающихся по результатам изучения следующих дисциплин: «Операционные системы», «Сети и телекоммуникации», «Сетевые технологии» и «Базы данных». Материал курсов необходим для понимания внутреннего устройства инструментов управления разработкой, принципов их конфигурирования и взаимодействия между собой и с пользователями.

Дисциплина «Управление производственным процессом разработки программного обеспечения» реализуется в 6 семестре в рамках обязательной части дисциплин (модулей) Блока 1 и является обязательной дисциплиной.

Дисциплина «Управление производственным процессом разработки программного обеспечения» направлена на формирование компетенций:

Способен определять круг задач в рамках поставленной цели и выбирать оптимальные способы их решения, исходя из действующих правовых норм, имеющихся ресурсов и ограничений (УК-2), в части следующих индикаторов достижения компетенции:

УК-2.1 Знать: виды ресурсов и ограничений для решения профессиональных задач; основные методы оценки разных способов решения задач; действующее законодательство и правовые нормы, регулирующие профессиональную деятельность

Способен осуществлять социальное взаимодействие и реализовывать свою роль в команде (УК-3), в части следующих индикаторов достижения компетенции:

УК-3.2 Уметь: устанавливать и поддерживать контакты, обеспечивающие успешную работу в коллективе; применять основные методы и нормы социального взаимодействия для реализации своей роли и взаимодействия внутри команды

УК-3.3 Владеть: простейшими методами и приемами социального взаимодействия и работы в команде

Способен участвовать в разработке стандартов, норм и правил, а также технической документации, связанной с профессиональной деятельностью (ОПК-4), в части следующих индикаторов достижения компетенции:

ОПК-4.1 Знать: основные стандарты оформления технической документации на различных стадиях жизненного цикла информационной системы

ОПК-4.2 Уметь: применять стандарты оформления технической документации на различных стадиях жизненного цикла информационной системы

ОПК-4.3 Владеть: навыками составления технической документации на различных этапах жизненного цикла информационной системы

Способен разрабатывать бизнес-планы и технические задания на оснащение отделов, лабораторий, офисов компьютерным и сетевым оборудованием (ОПК-6), в части следующих индикаторов достижения компетенции:

ОПК-6.1 Знать: принципы формирования и структуру бизнес-планов и технических заданий на оснащение отделов, лабораторий, офисов компьютерным и сетевым оборудованием

ОПК-6.2 Уметь: проводить анализ потребностей подразделений в оснащении компьютерным и сетевым оборудованием и составлять бизнес-планы и технические задания на оснащение подразделений.

ОПК-6.3 Владеть: навыками разработки технических заданий

Способен участвовать в настройке и наладке программно-аппаратных комплексов (ОПК-7), в части следующих индикаторов достижения компетенции:

ОПК-7.1 Знать: методы настройки, наладки программно-аппаратных комплексов

ОПК-7.2 Уметь: анализировать техническую документацию, производить настройку, наладку и тестирование программно-аппаратных комплексов

Перечень основных разделов дисциплины

Преподавание дисциплины предусматривает проведение следующих видов учебных занятий:

- лекции
- практические занятия, на которых студент знакомится с инструментарием и участвует в разработке учебного проекта
- самостоятельная работа студента над учебным проектом

На лекциях рассказывается о современном процессе разработки программного обеспечения и с какими задачами сталкивается проектная команда. Объясняются современные способы решения задач управления разработкой и разбираются основные инструменты, применяемые в коммерческой разработке.

На практических занятиях студенты знакомятся с основными инструментами разработки и используют их при решении простых задач.

В течении всего курса студенты выполняют проект в командах по 2-5 человек. В процессе выполнения проекта студенты должны согласовать командный процесс разработки, настроить инструменты разработки и выполнить проект, используя настроенный процесс. В конце курса студенты презентуют выполненный проект и созданный инструментарий для его выполнения.

Общий объем дисциплины – 3 зачетные единицы (108 часов).

Правила аттестации по дисциплине.

Текущий контроль по дисциплине «Управление производственным процессом разработки программного обеспечения» осуществляется на практических занятиях. На практических занятиях студенты знакомятся с инструментами разработки программного обеспечения и используют их для решения задач, выданных преподавателем. Кроме того, на практических занятиях осуществляется контроль выполнения проекта командами, проверяется степень готовности процесса разработки, инструментов разработки и самого проекта.

Промежуточная аттестация по дисциплине «Управление производственным процессом разработки программного обеспечения» проводится по завершению каждого периода ее освоения (6-й семестр). Результаты промежуточной аттестации по дисциплине оцениваются по шкале «неудовлетворительно», «удовлетворительно», «хорошо», «отлично». Оценки «отлично», «хорошо», «удовлетворительно», означают успешное прохождение промежуточной аттестации.

В 6-ом семестре оценка за освоение дисциплины выставляется при проведении дифференциального зачета, на котором учитываются следующие факторы:

1. Результат презентации проекта, выполненного студентом в команде из 2-5 человек
2. Общее понимание студентом процесса разработки и области применения тех или иных инструментов разработки, показанного на собеседовании

Учебно-методическое обеспечение дисциплины.

Учебно-методический комплекс по дисциплине «Управление производственным процессом разработки программного обеспечения» в электронной информационно-образовательной среде НГУ: <http://ccfit.nsu.ru/~den/>

1. Внешние требования к дисциплине

Таблица 1.1

Компетенция УК-2 Способен определять круг задач в рамках поставленной цели и выбирать оптимальные способы их решения, исходя из действующих правовых норм, имеющихся ресурсов и ограничений, в части следующих индикаторов достижения компетенции:
УК-2.1 Знать: виды ресурсов и ограничений для решения профессиональных задач; основные методы оценки разных способов решения задач; действующее законодательство и правовые нормы, регулирующие профессиональную деятельность
Компетенция УК-3 Способен осуществлять социальное взаимодействие и реализовывать свою роль в команде, в части следующих индикаторов достижения компетенции:
УК-3.2 Уметь: устанавливать и поддерживать контакты, обеспечивающие успешную работу в коллективе; применять основные методы и нормы социального взаимодействия для реализации своей роли и взаимодействия внутри команды
УК-3.3 Владеть: простейшими методами и приемами социального взаимодействия и работы в команде
Компетенция ОПК-4 Способен участвовать в разработке стандартов, норм и правил, а также технической документации, связанной с профессиональной деятельностью, в части следующих индикаторов достижения компетенции:
ОПК-4.1 Знать: основные стандарты оформления технической документации на различных стадиях жизненного цикла информационной системы
ОПК-4.2 Уметь: применять стандарты оформления технической документации на различных стадиях жизненного цикла информационной системы
ОПК-4.3 Владеть: навыками составления технической документации на различных этапах жизненного цикла информационной системы
Компетенция ОПК-6 Способен разрабатывать бизнес-планы и технические задания на оснащение отделов, лабораторий, офисов компьютерным и сетевым оборудованием, в части следующих индикаторов достижения компетенции:
ОПК-6.1 Знать: принципы формирования и структуру бизнес-планов и технических заданий на оснащение отделов, лабораторий, офисов компьютерным и сетевым оборудованием
ОПК-6.2 Уметь: проводить анализ потребностей подразделений в оснащении компьютерным и сетевым оборудованием и составлять бизнес-планы и технические задания на оснащение подразделений
ОПК-6.3 Владеть: навыками разработки технических заданий
Компетенция ОПК-7 Способен участвовать в настройке и наладке программно-аппаратных комплексов, в части следующих индикаторов достижения компетенции:
ОПК-7.1 Знать: методы настройки, наладки программно-аппаратных комплексов
ОПК-7.2 Уметь: анализировать техническую документацию, производить настройку, наладку и тестирование программно-аппаратных комплексов

2. Требования к результатам освоения дисциплины

Таблица 2.1

Результаты изучения дисциплины по уровням освоения (иметь представление, знать, уметь, владеть)	Формы организации занятий		
	Лекции	Практики / семинары	Самостоятельная работа
УК-2.1 Знать: виды ресурсов и ограничений для решения профессиональных задач; основные методы оценки разных способов решения задач; действующее законодательство и правовые нормы, регулирующие профессиональную деятельность			

1. Знать принципы решения задач и проблем, возникающих при разработке ПО; как применяются современные инструменты для решения возникающих задач и проблем	+	+	+
2. Знать: ограничения и область применимости инструментов командной разработки ПО	+	+	+
УК-3.2 Уметь: устанавливать и поддерживать контакты, обеспечивающие успешную работу в коллективе; применять основные методы и нормы социального взаимодействия для реализации своей роли и взаимодействия внутри команды			
3. Уметь выполнять поставленные задачи в коллективе разработчиков;	+	+	+
УК-3.3 Владеть: простейшими методами и приемами социального взаимодействия и работы в команде			
4. Владеть навыками пользования комбинированной средой управления проектами Redmine для организации работы команды разработчиков	+	+	+
ОПК-4.1 Знать: основные стандарты оформления технической документации на различных стадиях жизненного цикла информационной системы			
5. Знать основные фазы процесса разработки ПО	+	+	+
ОПК-4.2 Уметь: применять стандарты оформления технической документации на различных стадиях жизненного цикла информационной системы			
6. Уметь использовать современные инструменты управления разработкой программного обеспечения	+	+	+
ОПК-4.3 Владеть: навыками составления технической документации на различных этапах жизненного цикла информационной системы			
7. Владеть навыками пользования комбинированной средой управления проектами Redmine для ведения полной документации по проекту	+	+	+
ОПК-6.1 Знать: принципы формирования и структуру бизнес-планов и технических заданий на оснащение отделов, лабораторий, офисов компьютерным и сетевым оборудованием			
8. Знать современные инструменты управления разработкой программного обеспечения	+	+	+
ОПК-6.2 Уметь: проводить анализ потребностей подразделений в оснащении компьютерным и сетевым оборудованием и составлять бизнес-планы и технические задания на оснащение подразделений			
9. Уметь выбирать инструменты разработки, подходящие к конкретному проекту; конфигурировать выбранные инструменты для нужд проекта	+	+	+
ОПК-6.3 Владеть: навыками разработки технических заданий			
10. Владеть навыками пользования комбинированной средой управления проектами Redmine для разработки технических заданий	+	+	+
ОПК-7.1 Знать: методы настройки, наладки программно-аппаратных комплексов			
11. Знать принципы работы СКВ Git, комбинированной среды управления проектами Redmine, системы непрерывной интеграции Jenkins	+	+	+
ОПК-7.2 Уметь: анализировать техническую документацию, производить настройку, наладку и			

тестирование программно-аппаратных комплексов			
12. Уметь устанавливать и настраивать: СКВ Git, комбинированную среду управления проектами Redmine, систему непрерывной интеграции Jenkins	+	+	+

3. Содержание и структура учебной дисциплины

Таблица 3.1

Темы лекций	Активные формы, час. (входит в общее кол-во часов)	Часы	Ссылки на результаты обучения
Семестр: 6			
1. Введение в предмет: <ul style="list-style-type: none"> • Задачи и проблемы, возникающие при разработке программного обеспечения • Инструменты управления разработкой программного обеспечения • Главный вопрос конфигурационного управления 	0	2	1, 2, 5, 8
2. Планирование и управление проектами <ul style="list-style-type: none"> • Планирование с использованием диаграммы Ганта • Методы оценивания задач • Современные методологии управления проектами: Agile, eXtreme Programming, Scrum, Lean, Kanban 	0	4	1, 2, 5, 8
<ul style="list-style-type: none"> • Коллективная разработка с использованием системы контроля версий (СКВ) • Базовые принципы работы СКВ • Классические системы контроля версий (СКВ). Subversion. • Распределенные СКВ. Git. • Сценарии использования распределенных СКВ. • Проблемы коллективной разработки с использованием распределенных СКВ: разрешение конфликтов (команда merge), ветви и теги (команды branch и tag) • Интеграция СКВ в проектную инфраструктуру • Развертывание СКВ • Системы именования ветвей и версий 	0	4	1, 2, 3, 5, 6, 8, 9, 11, 12
3. Багтрекеры	0	4	1, 2, 3, 4, 5, 6,

<ul style="list-style-type: none"> • Необходимость багтрекеров • История появления багтрекеров • Жизненный цикл бага • Современные багтрекеры: JIRA и Redmine • Вариации жизненного цикла задачи 			7, 8, 9, 10, 11, 12
<p>4. Автоматизированная сборка продукта</p> <ul style="list-style-type: none"> • Необходимость автоматизации сборки • Системы сборки первого поколения. Make, rake, cake, SBT. • Первые декларативные системы сборки. Apache Ant • Дальнейшее развитие декларативных систем сборки: Apache Maven и Gradle. 	0	2	1, 2, 3, 5, 6, 8, 9
<p>5. Непрерывная интеграция (Continuous Integration)</p> <ul style="list-style-type: none"> • Системы непрерывной интеграции на примере Jenkins • Архитектура инструментов для непрерывной интеграции • Непрерывная интеграция большого проекта • Сложности непрерывной интеграции • Другие инструменты непрерывной интеграции 	0	4	1, 2, 3, 5, 6, 8, 9, 11, 12
<p>6. Тестирование (QA)</p> <ul style="list-style-type: none"> • Принципы модульного тестирования (Unit testing) • Библиотеки для модульного тестирования • Требования к модульным тестам • Анализ тестового покрытия (code coverage) • Распространенные подходы к модульному тестированию. Test Driven Development • Дальнейшее развитие идеи модульных тестов. Behavior Driven Development 	0	4	1, 2, 3, 5, 6, 8, 9
<p>7. Нагрузочное тестирование</p> <ul style="list-style-type: none"> • Роль нагрузочного тестирования • Принципы и сценарии нагрузочного тестирования • Инструменты для нагрузочного тестирования (ab, JMeter) 	0	2	1, 2, 3, 5, 6, 8, 9
<p>8. Управление знаниями</p> <ul style="list-style-type: none"> • Необходимость управления знаниями • Способы организации знаний (списки часто задаваемых вопросов (FAQ), HOWTO) • Использование wiki для организации знаний • Генераторы документации на примере doxygen и javadoc 	0	2	1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12
<p>9. Автоматизация технической поддержки (Trouble ticket systems)</p>	0	2	1, 2, 3, 5, 6, 8, 9

<ul style="list-style-type: none"> • Организация взаимодействия с пользователями • Техническая поддержка по электронной почте • Инструменты автоматизации технической поддержки 			
<p>10. Управление конфигурацией</p> <ul style="list-style-type: none"> • Основные инструменты управления конфигурацией: Ansible, Puppet, Salt, Chef • Push и Pull модели управления • Примеры управления конфигурацией с использованием Ansible 	0	2	1, 2, 3, 5, 6, 8, 9
Итого:		32	

Таблица 3.2

Темы практических занятий	Активные формы, час. (входит в общее кол-во часов)	Часы	Ссылки на результаты обучения	Учебная деятельность
Семестр: 6				
1. Формирование команд	2	2	1,2	Формирование проектных команд, распределение ролей в команде, обсуждение возможных тем проекта
2. Знакомство с процессом и инструментами разработки	2	2	1, 2, 3, 4, 6, 7, 9, 10, 11, 12	Утверждение темы проекта, процесса и инструментов разработки, состава проектных команд
3. Установка и настройка системы контроля версий (СКВ)	2	2	4, 5, 7, 8, 11, 12	Разработка структуры проекта в СКВ. Обсуждение и утверждение правил написания и комментирования кода. Обсуждение процедур работы с СКВ, проверки кода и именованя веток.
4. Использование системы контроля версий (СКВ)	4	4	2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12	Выполнение поставленных задач по программированию, участие в командных совещаниях, работа с СКВ, обсуждение возникающих вопросов с преподавателем, контроль сроков и рисков проекта
5. Настройка Redmine под утвержденный	2	2	2, 3, 4, 5, 6, 7,	Утверждение процесса разработки и правил

процесс разработки			8, 9, 10, 11, 12	создания задач в Redmine. Регистрация текущих задач. Обсуждение и утверждение правил совместной работы в Redmine и СКВ.
6. Использование Redmine	4	4	2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12	Выполнение поставленных задач по программированию, участие в командных совещаниях, работа с СКВ и Redmine, обсуждение возникающих вопросов с преподавателем, контроль сроков и рисков проекта.
7. Настройка и использование Jenkins	4	4	2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12	Настройка автоматической сборки с использованием системы Jenkins.
8. Настройка и использование вспомогательных инструментов разработки	4	4	2, 3, 4, 5, 6, 7, 8, 9, 10	Обсуждение и утверждение вспомогательных инструментов разработки. Настройка новых инструментов.
9. Защита выполненных проектов	8	8	1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12	Защита выполненных проектов. Презентация разработанной структуры кода в СКВ, процесса разработки в Redmine, автоматической сборки/развертки в Jenkins и используемых вспомогательных инструментов.
Итого:	32	32		

4. Самостоятельная работа студентов

Таблица 4.1

№	Виды самостоятельной работы	Ссылки на результаты обучения	Часы на выполнение	Часы на консультации
Семестр: 6				
1	Участие в командной работе над проектом	1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12	30	2
	Работа над проектом в составе команды, использование изученных продуктов управления разработкой, применение изученных процессов разработки.			
2	Подготовка к презентации проекта и к дифференцированному зачету	1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12	10	0
	Подготовка к презентации проекта и дифференцированному зачету на основе			

изученных тем.			
Итого:		40	2

5. Образовательные технологии

В ходе реализации учебного процесса по дисциплине проводятся лекционные и семинарские занятия. Темы, рассматриваемые на лекциях и изучаемые самостоятельно, закрепляются на семинарах, по вопросам, вызывающим затруднения, проводятся консультации. Применяются такие формы проведения практических занятий, как дискуссии, обсуждение и защита результатов работы, а также применяются следующие интерактивные формы обучения (таблица 5.1).

Таблица 5.1

1	Выполнение проекта	УК-2.1, УК-3.2, УК3.3, ОПК-4.1, ОПК-4.2, ОПК-4.3, ОПК-6.1, ОПК-6.2, ОПК-6.3, ОПК-7.1, ОПК-7.2
Формируемые умения: 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12		
Краткое описание применения: Разработка программного продукта в команде. Настройка процесса разработки и средств разработки программного обеспечения. Презентация сконфигурированного процесса и инфраструктуры, показ как использовалась готовая инфраструктура при разработке программного продукта.		

Для организации и контроля самостоятельной работы студентов, а также проведения консультаций применяются информационно-коммуникационные технологии (таблица 5.2).

Таблица 5.2

Информирование	Страница с информацией о курсе: http://ccfit.nsu.ru/~den/ Рассылка сообщений по адресам электронной почты обучающихся
Консультирование	Прием вопросов по электронной почте d.anoykin@g.nsu.ru
Контроль	Страница с информацией о курсе: http://ccfit.nsu.ru/~den/
Размещение учебных материалов	Страница с информацией о курсе: http://ccfit.nsu.ru/~den/

6. Правила аттестации студентов по учебной дисциплине

Текущий контроль по дисциплине «Управление производственным процессом разработки программного обеспечения» осуществляется на практических занятиях. На практических занятиях студенты знакомятся с инструментами разработки программного обеспечения и используют их для решения задач, выданных преподавателем. Кроме того, на практических занятиях осуществляется контроль выполнения проекта командами,

проверяется степень готовности процесса разработки, инструментов разработки и самого проекта.

Промежуточная аттестация по дисциплине «Управление производственным процессом разработки программного обеспечения» проводится по завершению каждого периода ее освоения (семестра). Результаты промежуточной аттестации по дисциплине оцениваются по шкале «неудовлетворительно», «удовлетворительно», «хорошо», «отлично». Оценки «отлично», «хорошо», «удовлетворительно», означают успешное прохождение промежуточной аттестации.

В 6-ом семестре оценка за освоение дисциплины выставляется при проведении дифференциального зачета, на котором учитываются следующие факторы:

1. Результат презентации проекта, выполненного студентом в команде из 2-5 человек
2. Общее понимание студентом процесса разработки и области применения тех или иных инструментов разработки, показанного на собеседовании

В 6 семестре результаты промежуточной аттестации определяются оценками «отлично», «хорошо», «удовлетворительно», «неудовлетворительно». Оценки «отлично», «хорошо», «удовлетворительно» означают успешное прохождение промежуточной аттестации.

Оценка «отлично» соответствует продвинутому уровню сформированности компетенции.

Оценка «хорошо» соответствует базовому уровню сформированности компетенции.

Оценка «удовлетворительно» соответствует пороговому уровню сформированности компетенции.

В таблице 6.1 представлено соответствие форм аттестации заявляемым требованиям к результатам освоения дисциплины.

Таблица 6.1

Коды компетенций ФГОС	Результаты обучения	Формы аттестации	
		Проект	Дифференцированный зачет
УК-2	УК-2.1 Знать: виды ресурсов и ограничений для решения профессиональных задач; основные методы оценки разных способов решения задач; действующее законодательство и правовые нормы, регулирующие профессиональную деятельность	+	+
УК-3	УК-3.2 Уметь: устанавливать и поддерживать контакты, обеспечивающие успешную работу в коллективе; применять основные методы и нормы социального взаимодействия для реализации своей роли и взаимодействия внутри команды	+	+
	УК-3.3 Владеть: простейшими методами и приемами социального взаимодействия и работы в команде	+	+
ОПК-4	ОПК-4.1 Знать: основные стандарты оформления технической документации на различных стадиях	+	+

	жизненного цикла информационной системы		
	ОПК-4.2 Уметь: применять стандарты оформления технической документации на различных стадиях жизненного цикла информационной системы	+	+
	ОПК-4.3 Владеть: навыками составления технической документации на различных этапах жизненного цикла информационной системы	+	+
ОПК-6	ОПК-6.1 Знать: принципы формирования и структуру бизнес-планов и технических заданий на оснащение отделов, лабораторий, офисов компьютерным и сетевым оборудованием	+	+
	ОПК-6.2 Уметь: проводить анализ потребностей подразделений в оснащении компьютерным и сетевым оборудованием и составлять бизнес-планы и технические задания на оснащение подразделений	+	+
	ОПК-6.3 Владеть: навыками разработки технических заданий	+	+
ОПК-7	ОПК-7.1 Знать: методы настройки, наладки программно-аппаратных комплексов	+	+
	ОПК-7.2 Уметь: анализировать техническую документацию, производить настройку, наладку и тестирование программно-аппаратных комплексов	+	+

Требования к структуре и содержанию проекта, оценочные средства, а также критерии оценки сформированности компетенций и освоения дисциплины в целом, представлены в Фонде оценочных средств, являющемся приложением 1 к настоящей рабочей программе дисциплины.

7. Литература

1. Зубкова, Т. М. Технология разработки программного обеспечения : учебное пособие / Т. М. Зубкова. — Санкт-Петербург : Лань, 2019. — 324 с. — ISBN 978-5-8114-3842-6. — Текст : электронный // Лань : электронно-библиотечная система. — URL: <https://e.lanbook.com/book/122176>
2. Новикова, Татьяна Сергеевна (д-р экон. наук). Управление проектами : практикум : [учебно-методическое пособие для вузов] / Т.С. Новикова, Н.В. Горбачева ; Федер. агентство по образованию, Новосиб. гос. ун-т, Экон. фак. Новосибирск : Редакционно-издательский центр НГУ, 2009. 28 с. ; 20 см. . (59 экз)
3. Загоруйко, Юрий Алексеевич. Инженерия знаний [Текст: электронный ресурс] : учебное пособие : [для студентов, изучающих курс "Инженерия знаний"] / Ю.А. Загоруйко, Г.Б. Загоруйко ; М-во образования и науки РФ, Новосиб. гос. ун-т, Фак. информ. технологий. (Новосибирск : РИЦ НГУ, 2016) . URL: <http://e-lib.nsu.ru/dsweb/Get/Resource-1052/page001.pdf> .

Интернет-ресурсы

Таблица 7.1

№ п/п	Наименование Интернет-ресурса	Краткое описание
1	https://git-scm.com/book/ru/v2	Git - fast version control

2	https://www.scrumguides.org/docs/scrumguide/v2017/2017-Scrum-Guide-Russian.pdf	Руководство по Скраму
3	https://riptutorial.com/ru/jenkins	Jenkins
4	https://www.redmine.org/projects/redmine/wiki/RusGuide	Redmine

8. Учебно-методическое и программное обеспечение дисциплины

8.1. Учебно-методическое обеспечение

Учебно-методический комплекс по дисциплине «Управление производственным процессом разработки программного обеспечения» в электронной информационно-образовательной среде НГУ: <http://ccfit.nsu.ru/~den/>

8.2. Программное обеспечение

Для обеспечения реализации дисциплины используется стандартный комплект программного обеспечения (ПО), включающий регулярно обновляемое лицензионное ПО Windows и MS Office.

Перечень специализированного программного обеспечения для изучения дисциплины представлен в таблице 8.1.

Специализированное программное обеспечение

Таблица 8.1

№	Наименование ПО	Назначение
1	IntelliJ IDEA Community Edition 2019	Среда разработки
2	Java SE Development Kit 8 (64-bit)	Среда разработки
3	Git v.2.18.0	Система контроля версий
4	Far Manager 3	Файловый менеджер
5	7-Zip v16.04	Файловый архиватор

9. Профессиональные базы данных и информационные справочные системы

1. Полнотекстовые журналы Springer Journals за 1997-2015 г., электронные книги (2005-2016 гг.), коллекция научных биомедицинских и биологических протоколов SpringerProtocols, коллекция научных материалов в области физических наук и инжиниринга SpringerMaterials, реферативная БД по чистой и прикладной математике zbMATH.

2. Электронная библиотека диссертаций Российской государственной библиотеки (ЭБД РГБ)

3. Электронные ресурсы Web of Science Core Collection (Thomson Reuters Scientific LLC.), Journal Citation Reports + ESI

4. Электронные БД JSTOR (США). Mathematics & Statistics.

10. Материально-техническое обеспечение

Таблица 10.1

№	Наименование	Назначение
1	Презентационное оборудование (мультимедиа-проектор, экран, компьютер для управления)	Для проведения лекционных занятий
2	Компьютерный класс (с выходом в Internet)	Для проведения практических занятий и организации самостоятельной работы

Материально-техническое обеспечение образовательного процесса по дисциплине для обучающихся из числа лиц с ограниченными возможностями здоровья осуществляется согласно «Порядку организации и осуществления образовательной деятельности по образовательным программам для инвалидов и лиц с ограниченными возможностями здоровья в Новосибирском государственном университете».

Министерство науки и высшего образования Российской Федерации
Федеральное государственное автономное образовательное учреждение
высшего образования «Новосибирский национальный исследовательский
государственный университет» (Новосибирский государственный университет, НГУ)

Факультет информационных технологий

СОГЛАСОВАНО

Декан ФИТ НГУ

М.М. Лаврентьев

«23» июля 2020 г.

ФОНД ОЦЕНОЧНЫХ СРЕДСТВ ПРОМЕЖУТОЧНОЙ АТТЕСТАЦИИ
по дисциплине **Управление производственным процессом разработки программного обеспечения**

Направление подготовки: 09.03.01 ИНФОРМАТИКА И ВЫЧИСЛИТЕЛЬНАЯ ТЕХНИКА

Направленность (профиль): Программная инженерия и компьютерные науки

Квалификация: бакалавр

Форма обучения: очная

Год обучения: 3, семестр 6

Форма аттестации	Семестр
Дифференцированный зачет	6

Новосибирск 2020

Фонд оценочных средств промежуточной аттестации по дисциплине является **Приложением 1** к рабочей программе дисциплины «Управление производственным процессом разработки программного обеспечения», реализуемой в рамках образовательной программы высшего образования – программы бакалавриата 09.03.01 Информатика и вычислительная техника, направленность (профиль): Программная инженерия и компьютерные науки

Фонд оценочных средств промежуточной аттестации по дисциплине утвержден решением ученого совета факультета информационных технологий, протокол № 77 от 22.07.2020.

Разработчики:

Старший преподаватель кафедры общей информатики ФИТ,
магистр математики

Д.А. Анойкин

Заведующий кафедрой общей информатики ФИТ,
доктор физико-математических наук

Д.Е. Пальчунов

Ответственный за образовательную программу:
доцент кафедры систем информатики ФИТ,
кандидат технических наук

А.А. Романенко

1. Содержание и порядок проведения промежуточной аттестации по дисциплине

1.1. Общая характеристика содержания промежуточной аттестации

Промежуточная аттестация по дисциплине «Управление производственным процессом разработки программного обеспечения» проводится по завершению периода освоения образовательной программы (семестра) для оценки сформированности компетенций в части следующих индикаторов достижения компетенции (таблица П1.1).

Таблица П1.1

Код	Компетенции, формируемые в рамках дисциплины «Управление производственным процессом разработки программного обеспечения»	Семестр 6	
		Проект	Дифференцированный зачет
УК-2	Способен определять круг задач в рамках поставленной цели и выбирать оптимальные способы их решения, исходя из действующих правовых норм, имеющихся ресурсов и ограничений		
УК-2.1	Знать: виды ресурсов и ограничений для решения профессиональных задач; основные методы оценки разных способов решения задач; действующее законодательство и правовые нормы, регулирующие профессиональную деятельность	+	+
УК-3	Способен осуществлять социальное взаимодействие и реализовывать свою роль в команде		
УК-3.2	Уметь: устанавливать и поддерживать контакты, обеспечивающие успешную работу в коллективе; применять основные методы и нормы социального взаимодействия для реализации своей роли и взаимодействия внутри команды	+	+
УК-3.3	Владеть: простейшими методами и приемами социального взаимодействия и работы в команде	+	+
ОПК-4	Способен участвовать в разработке стандартов, норм и правил, а также технической документации, связанной с профессиональной деятельностью		
ОПК-4.1	Знать: основные стандарты оформления технической документации на различных стадиях жизненного цикла информационной системы	+	+
ОПК-4.2	Уметь: применять стандарты оформления технической документации на различных стадиях жизненного цикла информационной системы	+	+
ОПК-4.3	Владеть: навыками составления технической документации на различных этапах	+	+

	жизненного цикла информационной системы		
ОПК-6	Способен разрабатывать бизнес-планы и технические задания на оснащение отделов, лабораторий, офисов компьютерным и сетевым оборудованием		
ОПК-6.1	Знать: принципы формирования и структуру бизнес-планов и технических заданий на оснащение отделов, лабораторий, офисов компьютерным и сетевым оборудованием	+	+
ОПК-6.2	Уметь: проводить анализ потребностей подразделений в оснащении компьютерным и сетевым оборудованием и составлять бизнес-планы и технические задания на оснащение подразделений	+	+
ОПК-6.3	Владеть: навыками разработки технических заданий	+	+
ОПК-7	Способен участвовать в настройке и наладке программно-аппаратных комплексов		
ОПК-7.1	Знать: методы настройки, наладки программно-аппаратных комплексов	+	+
ОПК-7.2	Уметь: анализировать техническую документацию, производить настройку, наладку и тестирование программно-аппаратных комплексов	+	+

Промежуточная аттестация включает 2 этапа. Все компетенции оцениваются по результатам выполнения и презентации проекта (внимание на практические навыки) и собеседования во время дифференцированного зачета (внимание на теоретические знания).

Тематика вопросов на собеседовании носит комплексный характер, т.к. включает вопросы ситуационно-производственного, практического, а также научно-исследовательского содержания.

1.2. Порядок проведения промежуточной аттестации по дисциплине

Промежуточная аттестация по дисциплине «Управление производственным процессом разработки программного обеспечения» проводится по завершению каждого периода ее освоения (семестра). Результаты промежуточной аттестации по дисциплине оцениваются по шкале «неудовлетворительно», «удовлетворительно», «хорошо», «отлично». Оценки «отлично», «хорошо», «удовлетворительно», означают успешное прохождение промежуточной аттестации.

Оценка за освоение дисциплины выставляется при проведении дифференциального зачета, на котором учитываются следующие факторы:

1. Работа над проектом и результат презентации проекта, выполненного студентом в команде из 2-5 человек

2. Общее понимание студентом процесса разработки и области применения тех или иных инструментов разработки, показанного на собеседовании

2. Требования к структуре и содержанию фонда оценочных средств промежуточной аттестации по дисциплине

Перечень оценочных средств, применяемых на каждом этапе проведения промежуточной аттестации по дисциплине, представлен в таблице П1.2.

Таблица П1.2

№ п/п	Наименование оценочного средства	Краткая характеристика оценочного средства	Представление оценочного средства в фонде
Семестр 6			
1 этап - проект			
1	Проект	Конечный продукт, получаемый в результате планирования и выполнения комплекса учебных и исследовательских заданий. Позволяет оценить умения обучающихся самостоятельно конструировать свои знания в процессе решения практических задач и проблем, ориентироваться в информационном пространстве и уровень сформированности аналитических, исследовательских навыков, навыков практического и творческого мышления. Может выполняться в индивидуальном порядке или группой обучающихся.	Требования к оформлению и презентации проекта
2 этап – дифференцированный зачет			
2	Собеседование	Средство контроля, организованное как специальная беседа преподавателя с обучающимся на темы, связанные с изучаемой дисциплиной, и рассчитанное на выяснение объема знаний обучающегося по определенному разделу, теме, проблеме и т.п.	Примеры вопросов

2.1 Требования к структуре и содержанию оценочных средств аттестации

2.1.1 Требования к оформлению и презентации проекта

В конце семестра происходит защита готовых учебных проектов проектными командами. В качестве результата работы команда предоставляет следующие артефакты:

1. Полностью сконфигурированный проект в системе Redmine:
 - a. Присутствует полное описание проекта, не менее 200 слов
 - b. Все участники проекта должны быть зарегистрированы, им должны быть прописаны роли.
 - c. Создан свой workflow для проекта с подробным описанием каждого шага.
 - d. Проведена интеграция проекта с СКВ Git или Subversion
2. В системе Redmine должны быть зарегистрированы задачи для каждого участника проекта. Задачи должны быть полностью выполнены и проведены по всем шагам созданного workflow. В каждой задаче должны быть зарегистрированы артефакты выполнения: ссылка на соответствующий commit в СКВ, файл, ссылка на созданную веб-страницу и т.д.
3. Для каждого проекта должен существовать хотя бы один вспомогательный инструмент разработки: база знаний, система непрерывной интеграции, система статического анализа кода и т.д.
4. Результат проекта презентуется преподавателю в виде готового приложения или сайта. Рассказывается созданный workflow, показывается список выполненных задач, объясняется структура папок в СКВ.

2.1.2 Примеры вопросов для собеседования

- Назовите основные этапы жизненного цикла программного продукта
- Назовите основные требования предъявляемые к отчету об ошибке
- Опишите основные недостатки централизованных СКВ
- Опишите различие между распределенными СКВ Git и Mercurial

3. Критерии оценки сформированности компетенций в рамках промежуточной аттестации по дисциплине

Таблица П1.5

Шифр компетенций	Структурные элементы оценочных средств	Показатель сформированности	Не сформирован (2 балла)	Пороговый уровень (3 балла)	Базовый уровень (4 балла)	Продвинутый (5 баллов)
УК-2	Проект Дифференцированный зачет	УК-2.1 Знать: виды ресурсов и ограничений для решения профессиональных задач; основные методы оценки разных способов решения задач; действующее законодательство и правовые нормы, регулирующие профессиональную деятельность	Может описать только одну задачу (или проблему), возникающую при разработке программного обеспечения. Не может привести пример использования СКВ, багтрекера или системы непрерывной интеграции.	Перечисляет несколько основных задач и проблем, возникающих при разработке программного обеспечения. Описывает принципы решения проблем, связанных с хранением и изменением программного кода. Приводит примеры использования систем контроля версий (СКВ) для решения данных проблем.	Перечисляет основные задачи и проблемы, возникающие при разработке программного обеспечения. Описывает принципы решения проблем, связанных с хранением и изменением программного кода. Описывает принципы решения проблем, связанных с контролем изменений. Приводит примеры использования систем контроля версий (СКВ) и багтрекеров для решения данных	Перечисляет основные задачи и проблемы, возникающие при разработке программного обеспечения. Описывает принципы решения основных проблем. Приводит примеры использования (СКВ) и багтрекеров для решения проблем. Описывает использование еще как минимум 2-х инструментов для решения основных проблем разработки.

					проблем.	
УК-3	Проект Дифференцированный зачет	УК-3.2 Уметь: устанавливать и поддерживать контакты, обеспечивающие успешную работу в коллективе; применять основные методы и нормы социального взаимодействия для реализации своей роли и взаимодействия внутри команды	Не выполнил ни одной задачи при разработке учебного проекта	Самостоятельно выполнил хотя бы одну задачу при разработке учебного проекта (есть код в СКР и правильно оформленная задача в багтрекере).	Самостоятельно выполнил не менее 3-х задач при разработке учебного проекта (есть код в СКР и правильно оформленная задача в багтрекере). Участвовал в разработке правил работы над учебным проектом.	Самостоятельно выполнил не менее 3-х задач при разработке учебного проекта (есть код в СКР и правильно оформленная задача в багтрекере). Ставил задачи, проверял работу других участников проекта. Активно участвовал в разработке правил работы над учебным проектом.
УК-3	Проект Дифференцированный зачет	УК-3.3 Владеть: простейшими методами и приемами социального взаимодействия и работы в команде	Не использовал Redmine для организации работы команды разработчиков при разработке учебного проекта	Использовал систему Redmine для организации работы команды разработчиков при разработке учебного проекта.	Использовал систему Redmine для организации работы команды разработчиков при разработке учебного проекта. Конфигурировал систему для нужд проекта.	Использовал систему Redmine для организации работы команды разработчиков при разработке учебного проекта. Конфигурировал систему для нужд проекта. Участвовал в формировании правил работы в рамках разработки учебного проекта (именование задач, правила

						перехода между состояниями в Redmine, и т.д.).
ОПК-4	Проект Проект Дифференцированный зачет	ОПК-4.1 Знать: основные стандарты оформления технической документации на различных стадиях жизненного цикла информационной системы	Имеет фрагментарное представление о жизненном цикле разработки ПО.	Дает определение основных этапов жизненного цикла разработки ПО, формулирует цели и задачи каждого этапа, демонстрирует владение профессиональной терминологией.	Распознает все этапы жизненного цикла разработки, способен выделить ключевые активности и отнести их к соответствующим этапам.	Демонстрирует целостное представление о жизненном цикле программного продукта. Подробно рассказывает о ключевых активностях на каждой фазе процесса разработки
ОПК-4	Проект Проект Дифференцированный зачет	ОПК-4.2 Уметь: применять стандарты оформления технической документации на различных стадиях жизненного цикла информационной системы	Имеет фрагментарные знания об основных инструментах управления разработкой программного обеспечения.	Использует системы контроля версий и багтрекеры для типовых рабочих задач.	Использует системы контроля версий и багтрекеры для типовых рабочих задач. Описывает принципы использования еще как минимум 2-х инструментов при разработке проекта.	Использует системы контроля версий, багтрекеры, системы непрерывной интеграции для типовых рабочих задач. Описывает принципы использования всех изученных инструментов при разработке проекта.
ОПК-4	Проект Проект Дифференцированный зачет	ОПК-4.3 Владеть: навыками составления технической	Не использовал Redmine для ведения полной документации по	Использовал систему Redmine ведения полной документации по	Использовал систему Redmine ведения полной документации по	Использовал систему Redmine для организации работы команды

	зачет	документации на различных этапах жизненного цикла информационной системы	проекту при разработке учебного проекта	проекту при разработке учебного проекта.	проекту при разработке учебного проекта. Конфигурировал систему для ведения документации.	разработчиков при разработке учебного проекта. Конфигурировал систему для ведения документации. Участвовал в формировании правил ведения документации и контролировал их применение.
ОПК-6	Проект Проект Дифференцированный зачет	ОПК-6.1 Знать: принципы формирования и структуру бизнес-планов и технических заданий на оснащение отделов, лабораторий, офисов компьютерным и сетевым оборудованием	Имеет фрагментарные знания об основных инструментах управления разработкой программного обеспечения.	Перечисляет основные инструменты управления разработкой программного обеспечения. Использует системы контроля версий и багтрекеры для типовых рабочих задач.	Перечисляет все изученные инструменты управления разработкой программного обеспечения. Использует системы контроля версий и багтрекеры для типовых рабочих задач. Описывает принципы использования еще как минимум 2-х инструментов при разработке проекта.	Перечисляет все изученные инструменты управления разработкой программного обеспечения. Использует системы контроля версий, багтрекеры, системы непрерывной интеграции для типовых рабочих задач. Описывает принципы использования всех изученных инструментов при разработке проекта.
ОПК-	Проект	ОПК-6.2 Уметь:	Имеет	Определяет	Самостоятельно	Оценивает

6	Проект Дифференцированный зачет	проводить анализ потребностей подразделений в оснащении компьютерным и сетевым оборудованием и составлять бизнес-планы и технические задания на оснащение подразделений	фрагментарные знания об основных инструментах и методологиях разработки, и используемые в конкретном. Не использует инструменты для типовых рабочих задач.	основные инструменты и методологии разработки, используемые в конкретном проекте и соотнести их с конкретными целями, поставленными перед проектом. Использует инструменты для типовых рабочих задач.	выбирает необходимые инструменты с учетом особенностей конкретного проекта. Обосновывает свои решения. Называет конкретные ограничения инструментов и учитывает эти ограничения в процессе принятия решений	правильность выбора инструментов и методологий с учетом конкретного проекта, обосновывает свои выводы. Владеет инструментами в рамках решения нестандартных задач; демонстрирует возможность разрешения нештатных ситуаций, произошедших при использовании инструмента.
ОПК-6	Проект Дифференцированный зачет	ОПК-6.3 Владеть: навыками разработки технических заданий	Не использовал Redmine для разработки технических заданий при работе над учебным проектом.	Использовал систему Redmine для разработки технических заданий при работе над учебным проектом.	Использовал систему Redmine для разработки технических заданий при работе над учебным проектом. Конфигурировал систему для разработки технических заданий.	Использовал систему Redmine для разработки технических заданий при работе над учебным проектом. Конфигурировал систему для разработки технических заданий. Участвовал в формировании правил разработки технических заданий и контролировал их выполнение.

ОПК-7	Проект Дифференцированный зачет	ОПК-7.1 Знать: методы настройки, наладки программно-аппаратных комплексов	Не знает принципы работы 2-х и более систем из списка: СКВ Git, Redmine, Jenkins	Знает принципы работы как минимум 2-х систем из списка: СКВ Git, Redmine, Jenkins	Знает принципы работы всех систем из списка: СКВ Git, Redmine, Jenkins	Знает принципы работы всех систем из списка: СКВ Git, Redmine, Jenkins Может сформулировать варианты использования каждой из систем для определенного проекта.
ОПК-7	Проект Дифференцированный зачет	ОПК-7.2 Уметь: анализировать техническую документацию, производить настройку, наладку и тестирование программно-аппаратных комплексов	Не умеет устанавливать и конфигурировать 2 и более систем из списка: СКВ Git, Redmine, Jenkins	Умеет устанавливать и конфигурировать как минимум 2 системы из списка: СКВ Git, Redmine, Jenkins	Умеет устанавливать и конфигурировать все системы из списка: СКВ Git, Redmine, Jenkins	Умеет устанавливать и конфигурировать все системы из списка: СКВ Git, Redmine, Jenkins Умеет конфигурировать данные системы под специфичные нужды проекта.

4. Критерии выставления оценок по результатам промежуточной аттестации по дисциплине

Результаты промежуточной аттестации в 6 семестре определяются оценками «отлично», «хорошо», «удовлетворительно», «неудовлетворительно». Оценки «отлично», «хорошо», «удовлетворительно» означают успешное прохождение промежуточной аттестации.

Оценка за курс выставляется во время собеседования при сдаче устного дифференцированного зачета. При этом учитывается работа над учебным проектом во время практических занятий.

Оценка «отлично» соответствует продвинутому уровню сформированности компетенции.

Оценка «хорошо» соответствует базовому уровню сформированности компетенции.

Оценка «удовлетворительно» соответствует пороговому уровню сформированности компетенции.

Оценка «неудовлетворительно» выставляется, если хотя бы одна компетенция не сформирована.