

**МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ
РОССИЙСКОЙ ФЕДЕРАЦИИ**
Федеральное государственное бюджетное образовательное учреждение
высшего образования
«Воронежский государственный технический университет»

УТВЕРЖДАЮ
Декан факультета информационных
технологий и компьютерной безопасности
_____/П.Ю. Гусев/
подпись
«31» августа 2021 г.

РАБОЧАЯ ПРОГРАММА
дисциплины

«Управление жизненным циклом программного обеспечения»

Направление подготовки 09.04.01 Информатика и вычислительная техника

Профиль Жизненный цикл изделий в едином информационном пространстве цифрового производства

Квалификация выпускника магистр

Нормативный период обучения 2 года / 2 года и 4 м.

Форма обучения очная / заочная

Год начала подготовки 2021

Автор программы

_____*А.А. Филимонова*_____/А.А. Филимонова/

Заведующий кафедрой

Компьютерных

интеллектуальных

технологий проектирования

_____*М.И. Чижев*_____/М.И. Чижев/

Руководитель ОПОП

_____*М.И. Чижев*_____/М.И. Чижев/

Воронеж 2021

1. ЦЕЛИ И ЗАДАЧИ ДИСЦИПЛИНЫ

1.1. Цели дисциплины

Изучение основ гибкого управления программным обеспечением

1.2. Задачи освоения дисциплины

- изучение методов создания, развертывания и анализа программных систем;
- получение практических навыков применения гибких методов проектирования программного обеспечения;
- изучение архитектуры и возможностей систем управления жизненным циклом программного обеспечения.

2. МЕСТО ДИСЦИПЛИНЫ В СТРУКТУРЕ ОПОП

Дисциплина «Управление жизненным циклом программного обеспечения» относится к дисциплинам обязательной части блока Б1.

3. ПЕРЕЧЕНЬ ПЛАНИРУЕМЫХ РЕЗУЛЬТАТОВ ОБУЧЕНИЯ ПО ДИСЦИПЛИНЕ

Процесс изучения дисциплины «Управление жизненным циклом программного обеспечения» направлен на формирование следующих компетенций:

УК-2 - Способен управлять проектом на всех этапах его жизненного цикла

ОПК-6 - Способен разрабатывать компоненты программно-аппаратных комплексов обработки информации и автоматизированного проектирования;

ОПК-8 - Способен осуществлять эффективное управление разработкой программных средств и проектов.

Компетенция	Результаты обучения, характеризующие сформированность компетенции
УК-2	знать основные методы управления проектами разработки программного обеспечения
	уметь планировать разработку программного обеспечения
	владеть организации командной разработки программного обеспечения
ОПК-6	знать архитектуру и возможности систем управления жизненным циклом программного обеспечения
	уметь создавать и планировать проекты в системе управления жизненным циклом программного обеспечения
	владеть навыками коллективной разработки программного обеспечения с использованием системы управления жизненным циклом

ОПК-8	знать основные этапы жизненного цикла программного обеспечения
	уметь осуществлять поддержку всех этапов жизненного цикла программного обеспечения
	владеть навыками автоматизации управления жизненным циклом программного обеспечения

4. ОБЪЕМ ДИСЦИПЛИНЫ

Общая трудоемкость дисциплины «Управление жизненным циклом программного обеспечения» составляет 6 з.е.

Распределение трудоемкости дисциплины по видам занятий

очная форма обучения

Виды учебной работы	Всего часов	Семестры
		3
Аудиторные занятия (всего)	72	72
В том числе:		
Лекции	18	18
Лабораторные работы (ЛР)	54	54
Самостоятельная работа	117	117
Часы на контроль	27	27
Виды промежуточной аттестации - экзамен	+	+
Общая трудоемкость:		
академические часы	216	216
зач.ед.	6	6

заочная форма обучения

Виды учебной работы	Всего часов	Семестры
		3
Аудиторные занятия (всего)	8	8
В том числе:		
Лекции	4	4
Лабораторные работы (ЛР)	4	4
Самостоятельная работа	199	199
Часы на контроль	9	9
Виды промежуточной аттестации - экзамен	+	+
Общая трудоемкость:		
академические часы	216	216
зач.ед.	6	6

5. СОДЕРЖАНИЕ ДИСЦИПЛИНЫ (МОДУЛЯ)

5.1 Содержание разделов дисциплины и распределение трудоемкости по видам занятий

очная форма обучения

№ п/п	Наименование темы	Содержание раздела	Лекц	Лаб. зан.	СРС	Всего, час
-------	-------------------	--------------------	------	-----------	-----	------------

1	Модели жизненного цикла программного обеспечения	Понятие жизненного цикла программного обеспечения. Стандартизация. Традиционные модели жизненного цикла программного продукта. Преимущества и недостатки. Области применения	6	12	38	56
2	Принципы разработки программного обеспечения	Гибкая разработка. Agile. Lean.	4	8	20	32
3	Основные методы разработки программного обеспечения	Scrum. Kanban. RUP (Rational Unified Process). DSDM (Dynamic Systems Development Model), RAD (Rapid Application Development), XP (Extreme Programming)	4	8	20	32
4	Автоматизация управления жизненным циклом программного обеспечения	Системы SLM (Software Life Management), ALM (Application Lifecycle Management)	4	26	39	69
Итого			18	54	117	189

заочная форма обучения

№ п/п	Наименование темы	Содержание раздела	Лекц	Лаб. зан.	СРС	Всего, час
1	Модели жизненного цикла программного обеспечения	Понятие жизненного цикла программного обеспечения. Стандартизация. Традиционные модели жизненного цикла программного продукта. Преимущества и недостатки. Области применения	2	-	49	51
2	Принципы разработки программного обеспечения	Гибкая разработка. Agile. Lean.	-	-	50	50
3	Основные методы разработки программного обеспечения	Scrum. Kanban. RUP (Rational Unified Process). DSDM (Dynamic Systems Development Model), RAD (Rapid Application Development), XP (Extreme Programming)	-	-	50	50
4	Автоматизация управления жизненным циклом программного обеспечения	Системы SLM (Software Life Management), ALM (Application Lifecycle Management)	2	4	50	56
Итого			4	4	199	207

5.2 Перечень лабораторных работ

1. Подготовка проекта разработки программного обеспечения
2. Первоначальная настройка системы управления программным обеспечением Jira
3. Настройка интеграции с системой управления версиями программного обеспечения
4. Создание и планирование баг-трекера в системе Jira
5. Управление баг-трекером в системе Jira
6. Создание и планирование SCRUM проекта в системе Jira
7. Управление SCRUM проектом в системе Jira
8. Создание и планирование Kanban проекта в системе Jira
9. Управление Kanban проектом в системе Jira
10. Работа с дорожной картой проекта
11. Использование Jira Query Language (JQL)
12. Автоматизация процессов в системе Jira

6. ПРИМЕРНАЯ ТЕМАТИКА КУРСОВЫХ ПРОЕКТОВ (РАБОТ) И КОНТРОЛЬНЫХ РАБОТ

В соответствии с учебным планом освоение дисциплины не предусматривает выполнение курсового проекта (работы) или контрольной работы.

7. ОЦЕНОЧНЫЕ МАТЕРИАЛЫ ДЛЯ ПРОВЕДЕНИЯ ПРОМЕЖУТОЧНОЙ АТТЕСТАЦИИ ОБУЧАЮЩИХСЯ ПО ДИСЦИПЛИНЕ

7.1. Описание показателей и критериев оценивания компетенций на различных этапах их формирования, описание шкал оценивания

7.1.1 Этап текущего контроля

Результаты текущего контроля знаний и межсессионной аттестации оцениваются по следующей системе:

«аттестован»;

«не аттестован».

Компетенция	Результаты обучения, характеризующие сформированность компетенции	Критерии оценивания	Аттестован	Не аттестован
УК-2	знать основные методы управления проектами разработки программного обеспечения	Допуск к лабораторным работам	Выполнение работ в срок, предусмотренный в рабочих программах	Невыполнение работ в срок, предусмотренный в рабочих программах
	уметь планировать разработку программного обеспечения	Защита лабораторных работ	Выполнение работ в срок, предусмотренный в рабочих программах	Невыполнение работ в срок, предусмотренный в рабочих программах
	владеть организацией командной разработки программного обеспечения	Защита лабораторных работ	Выполнение работ в срок, предусмотренный в рабочих программах	Невыполнение работ в срок, предусмотренный в рабочих программах
ОПК-6	знать архитектуру и возможности систем управления жизненным циклом программного обеспечения	Допуск к лабораторным работам	Выполнение работ в срок, предусмотренный в рабочих программах	Невыполнение работ в срок, предусмотренный в рабочих программах
	уметь создавать и планировать проекты в системе управления жизненным циклом программного обеспечения	Защита лабораторных работ	Выполнение работ в срок, предусмотренный в рабочих программах	Невыполнение работ в срок, предусмотренный в рабочих программах
	владеть навыками коллективной разработки программного обеспечения с использованием системы управления жизненным циклом	Защита лабораторных работ	Выполнение работ в срок, предусмотренный в рабочих программах	Невыполнение работ в срок, предусмотренный в рабочих программах
ОПК-8	знать основные этапы жизненного цикла программного обеспечения	Допуск к лабораторным работам	Выполнение работ в срок, предусмотренный в рабочих программах	Невыполнение работ в срок, предусмотренный в рабочих программах

уметь осуществлять поддержку всех этапов жизненного цикла программного обеспечения	Защита лабораторных работ	Выполнение работ в срок, предусмотренный в рабочих программах	Невыполнение работ в срок, предусмотренный в рабочих программах
владеть навыками автоматизации управления жизненным циклом программного обеспечения	Защита лабораторных работ	Выполнение работ в срок, предусмотренный в рабочих программах	Невыполнение работ в срок, предусмотренный в рабочих программах

7.1.2 Этап промежуточного контроля знаний

Результаты промежуточного контроля знаний оцениваются в 3 семестре для очной формы обучения, 3 семестре для заочной формы обучения по четырехбалльной системе:

«отлично»;

«хорошо»;

«удовлетворительно»;

«неудовлетворительно».

Компетенция	Результаты обучения, характеризующие сформированность компетенции	Критерии оценивания	Отлично	Хорошо	Удовл.	Неудовл.
УК-2	знать основные методы управления проектами разработки программного обеспечения	Тест	Выполнение теста на 90-100%	Выполнение теста на 80-90%	Выполнение теста на 70-80%	В тесте менее 70% правильных ответов
	уметь планировать разработку программного обеспечения	Решение стандартных практических задач	Задачи решены в полном объеме и получены верные ответы	Продемонстрирован верный ход решения всех, но не получен верный ответ во всех задачах	Продемонстрирован верный ход решения в большинстве задач	Задачи не решены
	владеть организацией командной разработки программного обеспечения	Решение прикладных задач в конкретной предметной области	Задачи решены в полном объеме и получены верные ответы	Продемонстрирован верный ход решения всех, но не получен верный ответ во всех задачах	Продемонстрирован верный ход решения в большинстве задач	Задачи не решены
ОПК-6	знать архитектуру и возможности систем управления жизненным циклом программного обеспечения	Тест	Выполнение теста на 90-100%	Выполнение теста на 80-90%	Выполнение теста на 70-80%	В тесте менее 70% правильных ответов
	уметь создавать и планировать проекты в системе	Решение стандартных практических задач	Задачи решены в полном объеме и	Продемонстрирован верный ход решения всех,	Продемонстрирован верный ход решения в	Задачи не решены

	управления жизненным циклом программного обеспечения		получены верные ответы	но не получен верный ответ во всех задачах	большинстве задач	
	владеть навыками коллективной разработки программного обеспечения с использованием системы управления жизненным циклом	Решение прикладных задач в конкретной предметной области	Задачи решены в полном объеме и получены верные ответы	Продемонстрирован верный ход решения всех, но не получен верный ответ во всех задачах	Продемонстрирован верный ход решения в большинстве задач	Задачи не решены
ОПК-8	знать основные этапы жизненного цикла программного обеспечения	Тест	Выполнение теста на 90-100%	Выполнение теста на 80-90%	Выполнение теста на 70-80%	В тесте менее 70% правильных ответов
	уметь осуществлять поддержку всех этапов жизненного цикла программного обеспечения	Решение стандартных практических задач	Задачи решены в полном объеме и получены верные ответы	Продемонстрирован верный ход решения всех, но не получен верный ответ во всех задачах	Продемонстрирован верный ход решения в большинстве задач	Задачи не решены
	владеть навыками автоматизации управления жизненным циклом программного обеспечения	Решение прикладных задач в конкретной предметной области	Задачи решены в полном объеме и получены верные ответы	Продемонстрирован верный ход решения всех, но не получен верный ответ во всех задачах	Продемонстрирован верный ход решения в большинстве задач	Задачи не решены

7.2 Примерный перечень оценочных средств (типовые контрольные задания или иные материалы, необходимые для оценки знаний, умений, навыков и (или) опыта деятельности)

7.2.1 Примерный перечень заданий для подготовки к тестированию

1. Какой роли нет в Scrum?

- владелец продукта
- команда разработки
- скрам-мастер
- менеджер (+)

2. К распределенным системам управлениями версиями относятся (укажите 2):

- CVS
- GIT (+)
- Mercurian (+)
- RCS

3. Наличие обратных связей между этапами жизненного цикла, определяющих возможность проведения проверок и корректировок проектируемого ПС на каждой стадии разработки, является основной особенностью ...

- каскадной модели жизненного цикла
- итерационной модели жизненного цикла (+)
- спиральной модели жизненного цикла ПО
- модели жизненного цикла UML

4. Выявлением и устранением ошибок только на стадии тестирования, которая может растянуться во времени или вообще никогда не завершиться, характеризуется ...

- каскадная модель жизненного цикла (+)
- итерационная модель жизненного цикла
- спиральная модель жизненного цикла ПО
- модель жизненного цикла UML

5. Функционирование программного продукта в соответствии с техническим заданием соответствует характеристике эксплуатационных требований ...

- правильность (+)
- универсальность
- надежность
- адаптируемость

6. ... - система принципов и способов организации процесса разработки программ.

- методология (+)
- модель
- стратегия
- жизненный цикл

7. ... - структура, определяющая последовательность выполнения и взаимосвязи процессов, действий и задач на протяжении ЖЦ.

- методология
- модель (+)
- стратегия
- система

8. Модель ЖЦ ПО включает в себя (укажите 3):

- стадии (+)
- результаты выполнения работ (+)
- ключевые события (+)
- документацию
- персонал

9. ... - часть процесса создания ПО, ограниченная определенными временными рамками и заканчивающаяся выпуском конкретного продукта, определяемого заданными требованиями.

- стадия (+)
- разработка
- внедрение
- задача

10. Какая модель предполагает разбиение жизненного цикла проекта на последовательность итераций, каждая из которых напоминает «мини-проект», включая все фазы жизненного цикла в применении к созданию меньших фрагментов функциональности, по сравнению с проектом, в целом?

- каскадная
- инкрементная
- итеративная (+)
- V-модель

7.2.2 Примерный перечень заданий для решения стандартных задач

1. ... - это совокупность свойств, характеризующих способность программного обеспечения удовлетворять потребностям пользователя в соответствии с предназначением.

- требование
- качество (+)
- цель
- задача

2. Требование «Разработка приложения должна проводиться на языке C#» является

- функциональным
- ограничением (+)
- бизнес-правилом
- системным

3. Требование «В системе должен быть реализован поиск статей по ключевым словам и предоставляемым тегам» является

- функциональным (+)
- ограничением
- бизнес-правилом
- бизнес-требованием

4. Требования не должны противоречить... (укажите 2)

- требованиям своего уровня иерархии (+)
- требованиям пользователей
- требованиям «родительского» уровня (+)

5. Кто назначает приоритеты требований

- пользователь
- заказчик (+)
- исполнитель

6. Набор функциональных и нефункциональных требований, которые разработчики обязались реализовать в определенной версии

- стандартная версия требований
- текущая версия требований
- профессиональная версия требований
- базовая версия требований (+)

7. Определите статус требования «Требование отложено до соответствующих вариантов тестирования»

- Предложено
- Проверено (+)
- Отклонено
- Согласовано

8. Определите статус требования «Требование предложено, но не запланировано для реализации ни в одной из будущих версий»

- Предложено
- Проверено
- Отклонено (-)
- Согласовано

9. Какой процесс является механизмом суммирования и фильтрации изменений?

- процесс планирования изменений
- процесс управления изменениями
- процесс внедрения изменений
- процесс контроля изменений (+)

10. К какой фазе можно отнести бета-версию продукта?

- фаза начала
- фаза уточнения
- фаза конструирования (+)
- фаза внедрения

7.2.3 Примерный перечень заданий для решения прикладных задач

1. Что такое github.com?

- визуальный клиент git
- драйвер для git
- веб-сервис для хостинга IT-проектов и их совместной разработки, основанный на git (+)

- сайт разработчиков системы git

2. Какая команда показывает состояния файлов в рабочем каталоге и индексе?

- git status (+)
- git stat
- git add
- git new

3. Какая команда используется для отмены изменений?

- git reset (+)
- git rollback
- git undo
- git exit

4. Какая команда связывается с удалённым репозиторием и забирает из него все изменения, которых у вас пока нет и сохраняет их локально?

- git init
- git pull
- git merge
- git fetch (+)

5. В системе Jira для добавления участников к проекту необходимо указать:

- адрес электронной почты (+)
- уникальный идентификатор
- выбрать из списка
- учетную запись github
- фамилию и имя

6. Для проведения маркетинговых кампаний используется продукт:

- Jira Software
- Jira Service Management
- Jira Work Management (+)
- Jira Agile

7. ... применяются командой для планирования больших объемов работы на несколько месяцев вперед на уровне эпика и в рамках одного проекта.

- дорожные карты (+)
- канбан доски
- сетевые графики
- бэклоги

8. ... это большой объем работы, который можно разбить на отдельные задания, связанные с поставкой возможности.

- задача
- проект
- эпик (+)
- доска

9. ... отслеживают события в Jira, такие как создание задачи или изменение значения некоего поля.

- триггеры (+)
- события
- действия
- условия

10. В системе Jira ... позволяют автоматически выполнять задания и вносить на сайте определенные изменения.

- триггеры
- события
- действия (+)
- условия

7.2.4 Примерный перечень вопросов для подготовки к зачету

Не предусмотрено учебным планом

7.2.5 Примерный перечень заданий для подготовки к экзамену

1. Жизненный цикл программного обеспечения.
2. Стандартизация жизненного цикла ПО.
3. Традиционные модели жизненного цикла программного продукта.
4. Преимущества и недостатки традиционных моделей ЖЦ ПО.
5. Области применения моделей ЖЦ ПО.
6. Гибкая разработка ПО.
7. Методология Agile.
8. Методология Lean.
9. Методология Scrum.
10. Методология Kanban.
11. Методология RUP (Rational Unified Process).
12. Методология DSDM (Dynamic Systems Development Model).
13. Методология RAD (Rapid Application Development)
14. Методология XP (Extreme Programming)
15. SLM и ALM системы. Основные возможности.
16. SLM и ALM системы. Типовая архитектура.
17. Средства коллективной разработки программного обеспечения.
18. Системы управления версиями ПО.

7.2.6. Методика выставления оценки при проведении промежуточной аттестации

Экзамен проводится по тест-билетам, каждый из которых содержит 20 вопросов. Каждый правильный ответ на вопрос в тесте оценивается 1 баллом. Максимальное количество набранных баллов – 20.

1. Оценка «Неудовлетворительно» ставится в случае, если студент набрал менее 6 баллов.

2. Оценка «Удовлетворительно» ставится в случае, если студент набрал от 6 до 10 баллов

3. Оценка «Хорошо» ставится в случае, если студент набрал от 11 до 15 баллов.

4. Оценка «Отлично» ставится, если студент набрал от 16 до 20 баллов.)

7.2.7 Паспорт оценочных материалов

№ п/п	Контролируемые разделы (темы) дисциплины	Код контролируемой компетенции	Наименование оценочного средства
1	Модели жизненного цикла программного обеспечения	УК-2, ОПК-6, ОПК-8	Тест, защита лабораторных работ
2	Принципы разработки программного обеспечения	УК-2, ОПК-6, ОПК-8	Тест, защита лабораторных работ
3	Основные методы разработки программного обеспечения	УК-2, ОПК-6, ОПК-8	Тест, защита лабораторных работ
4	Автоматизация управления жизненным циклом программного обеспечения	УК-2, ОПК-6, ОПК-8	Тест, защита лабораторных работ

7.3. Методические материалы, определяющие процедуры оценивания знаний, умений, навыков и (или) опыта деятельности

Тестирование осуществляется, либо при помощи компьютерной системы тестирования, либо с использованием выданных тест-заданий на бумажном носителе. Время тестирования 30 мин. Затем осуществляется проверка теста экзаменатором и выставляется оценка согласно методике выставления оценки при проведении промежуточной аттестации.

Решение стандартных задач осуществляется, либо при помощи компьютерной системы тестирования, либо с использованием выданных задач на бумажном носителе. Время решения задач 30 мин. Затем осуществляется проверка решения задач экзаменатором и выставляется оценка, согласно методике выставления оценки при проведении промежуточной аттестации.

Решение прикладных задач осуществляется, либо при помощи компьютерной системы тестирования, либо с использованием выданных задач на бумажном носителе. Время решения задач 30 мин. Затем осуществляется проверка решения задач экзаменатором и выставляется оценка, согласно методике выставления оценки при проведении промежуточной аттестации.

8 УЧЕБНО МЕТОДИЧЕСКОЕ И ИНФОРМАЦИОННОЕ ОБЕСПЕЧЕНИЕ ДИСЦИПЛИНЫ)

8.1 Перечень учебной литературы, необходимой для освоения дисциплины

1. Аппело, Ю. Agile-менеджмент. Лидерство и управление командами=Managment 3.0. Leading Agile Developers, Developing Agile Leaders : [12+] / Ю. Аппело ; науч. ред. А. Обухова ; ред. А. Черникова ; пер. с англ. А. Олейник. – Москва : Альпина Паблишер, 2018. – 534 с. – Режим доступа: по подписке. – URL: <https://biblioclub.ru/index.php?page=book&id=570321>. – Библиогр. в кн. – ISBN 978-5-9614-6361-3. – Текст : электронный.

2. Лауферман, О. В. Разработка программного продукта: профессиональные стандарты, жизненный цикл, командная работа : учебное пособие / О. В. Лауферман, Н. И. Лыгина. — Новосибирск : Новосибирский государственный технический университет, 2019. — 75 с. — ISBN 978-5-7782-3893-0. — Текст : электронный // Электронно-библиотечная система IPR BOOKS : [сайт]. — URL: <https://www.iprbookshop.ru/99215.html>. — Режим доступа: для авторизир. пользователей

3. Организация самостоятельной работы обучающихся: методические указания для студентов, осваивающих основные образовательные программы высшего образования — бакалавриата, специалитета, магистратуры: методические указания / сост. В.Н. Почечихина, И.Н. Крючкова, Е.И. Головина, В.Р. Демидов; ФГБОУ ВО «Воронежский государственный технический университет». – Воронеж, 2020. – 14 с.

8.2 Перечень информационных технологий, используемых при осуществлении образовательного процесса по дисциплине, включая перечень лицензионного программного обеспечения, ресурсов информационно-телекоммуникационной сети «Интернет», современных профессиональных баз данных и информационных справочных систем:

Лицензионное ПО:

- Windows Professional 8.1 (7 и 8) Single Upgrade MVL A Each Academic (многопользовательская лицензия)
- Microsoft Office Word 2007
- Microsoft Office Power Point 2007
- IntelliJ IDEA

Свободно распространяемое ПО:

- Adobe Acrobat Reader
- Visual Studio Community Edition

Отечественное ПО:

- Яндекс.Браузер
- Архиватор 7z

Ресурс информационно-телекоммуникационной сети «Интернет»:

- Образовательный портал ВГТУ
- <http://www.edu.ru/>
- <https://www.atlassian.com/>
- <https://github.com/>
- <https://app.diagrams.net/>

Информационно-справочные системы:

- <http://window.edu.ru>
- <https://wiki.cchgeu.ru/>

Современные профессиональные базы данных:

- <https://proglib.io>

9 МАТЕРИАЛЬНО-ТЕХНИЧЕСКАЯ БАЗА, НЕОБХОДИМАЯ ДЛЯ ОСУЩЕСТВЛЕНИЯ ОБРАЗОВАТЕЛЬНОГО ПРОЦЕССА

Для проведения лекционных занятий необходима аудитория, оснащенная оборудованием для лекционных демонстраций и проекционной аппаратурой.

Для проведения лабораторных работ необходима лаборатория с ПК, оснащенными программами для проведения лабораторного практикума и обеспечивающими возможность доступа к локальной сети кафедры и Интернет, из следующего перечня:

- 202/2
- 213/2
- 215/2

Аудитории располагаются по адресу: г.Воронеж, ул.Плехановская, 11

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

По дисциплине «Управление жизненным циклом программного обеспечения» читаются лекции, проводятся лабораторные работы.

Основой изучения дисциплины являются лекции, на которых излагаются наиболее существенные и трудные вопросы, а также вопросы, не нашедшие отражения в учебной литературе.

Лабораторные работы выполняются на лабораторном оборудовании в соответствии с методиками, приведенными в указаниях к выполнению работ.

Вид учебных занятий	Деятельность студента
Лекция	Написание конспекта лекций: кратко, схематично, последовательно фиксировать основные положения, выводы, формулировки, обобщения; помечать важные мысли, выделять ключевые слова, термины. Проверка терминов, понятий с помощью энциклопедий, словарей, справочников с выписыванием толкований в тетрадь. Обозначение вопросов, терминов, материала, которые вызывают трудности, поиск ответов в рекомендуемой литературе. Если самостоятельно не удастся разобраться в материале, необходимо сформулировать вопрос и задать преподавателю на лекции или на практическом занятии.
Лабораторная работа	Лабораторные работы позволяют научиться применять теоретические знания, полученные на лекции при решении конкретных задач. Чтобы наиболее рационально и полно использовать все возможности лабораторных для подготовки к ним необходимо: следует разобрать лекцию по соответствующей теме, ознакомиться с соответствующим

	разделом учебника, проработать дополнительную литературу и источники, решить задачи и выполнить другие письменные задания.
Самостоятельная работа	Самостоятельная работа студентов способствует глубокому усвоения учебного материала и развитию навыков самообразования. Самостоятельная работа предполагает следующие составляющие: - работа с текстами: учебниками, справочниками, дополнительной литературой, а также проработка конспектов лекций; - подготовка к лабораторным работам; - работа над темами для самостоятельного изучения; - участие в работе студенческих научных конференций, олимпиад; - подготовка к промежуточной аттестации.
Подготовка к промежуточной аттестации	Готовиться к промежуточной аттестации следует систематически, в течение всего семестра. Интенсивная подготовка должна начаться не позднее, чем за месяц-полтора до промежуточной аттестации. Данные перед экзаменом, экзаменом три дня эффективнее всего использовать для повторения и систематизации материала.